PERSONALNI

računari

BIGZ 1831

81

APRIL 1992.

CENA 800 D

UPOREDNI TEST
LOTUS - QUATTRO PRO - EXCELL

UPOREDNI TEST
MEGA EVA - ATI WONDER - SPEEDSTAR

SOFTVER
QUICK C FOR WINDOWS

VISUAL BASIC

386MAX 6.0

KAKO DA NAPIŠEM...
...PROGRAM ZA POSLOVANJE PREDUZEĆA

NOVE TEHNOLOGIJE
ELEKTRONSKA FOTOGRAFIJA

SKENERI
HOWTEK SCANMASTER

OPERATIVNI SISTEMI
OS/2 2.0



YU ISSN 0352 7271

PERSONALNI
računari 81

časopis za informatiku
i računarstvo
YU ISSN 0352-7221
Izdaje BIGZ
april 1992 cena 800 din

Izdaje i štampa
Beogradski izdavačko-grafički zavod
11000 Beograd
Bulevar Vojvode Mišića 17
Generalni direktor
Ilija Rapajić
Direktor Novinskog sektora
Zoran Nikodijević
Glavni i odgovorni urednik
Jovan Regasek
Zamenik gl i odg urednika
Esad Jakupović
Tehnički urednik / sekretar redakcije
Natalija Desenčić
Urednik za hardver
Dejan V Veselinović
Stručna redakcija
Vesna Ćosić (aktuelnosti) Nenad Beločanin (baze podataka) Dejan Ristanović (softver) Jovan Skuljan (programiranje) Vladimir Stamenović (radne tabele) Dejan Veselinović (hardver) Dragutin Vuković (mreže) Zoran Životić (softver)
Adresa redakcije
11000 Beograd
Bulevar vojvode Mišića 17/III
Telefoni
653 748 (redakcija)
650-528 (prodaja)
Agencija BIGZ-a
(011) 651 793 653-565
Teleks
11855 BIGZ YU
Telefaks
(011) 648-140 647-955
Pretplata za zemlju
Za šest meseci 4800 (6 brojeva)
Za jednu godinu 8800 (11 brojeva)
(na žiro-račun D P BIGZ, 60802 603-23264)
Pretplata za inostranstvo
Za jednu godinu 51 USD 80 DEM 67 CHS 27 GBP 271 FRF
(na devizni račun D P BIGZ 60811-620 16101-820701 999-03377)
Zbog usporenog prenošenja uplata preko banke molimo pretplatnike da nam pošle svake nove uplate odmah pošalju foto kopiju uplatnice
Rukopisi se ne vraćaju

SEZAM
sistem za modemske komunikacije
(011) 648-899 (deset linija)
Upravnik sistema
Jovan Regasek
Administratori na sistemu
Zoran Životić i Dejan Ristanović
Moderatori
Dejan Balinda (CIVILIZACIJA), Jovan Bulajić (PC USER) Aleksandar Damnjanović (PC OS) Slobodan Kelezić (ORKA) Zoran Kehler (MOST) Vladimir Kostić (PC SOFT) Vladimir Krstonošić (VICEVI) Esad Jakupović (RAČUNARI) Dalibor Lanik (ATARI ST) Pavle Peković (PC PROG) Dejan Ristanović (NOVOSTI i KOMUNIKACIJE) Bojan Tepavčević (FORUM) Dejan Veselinović (PC HARD) i Zoran Životić (SEZAM)
Sekretarica sistema
Natalija Desenčić
Softver
Zoran Životić (glavne aplikacija) i Dejan Ristanović (eksterni programi)
Hardver
Server COMPAQ DESKPRO 386/25 3 MB RAM-a tvrdi disk 300 MB (COMPAQ Minhen) Mreža dve stanice COMPAQ 286N/12 1 MB RAM-a (COMPAQ Minhen) tri stanice „Ultra" AT 12 MHz D-LINK DE 100 LanSmart ("Ultra" Skopje) Modemi VIVa (Thelama Communications Beograd)

SADRŽAJ



Scena sa manekenkom Sanjom Đorđević (26 godina) koja je "majcinski" obgrlila sliku na monitoru verovatno može da se tumači na različite načine Ali bez obzira šta su stvarno imali na umu autori fotografije, Studio Tešić i Nenad Petrović, ona, pre svega, ilustruje fantastične domete najnovijih kolor skenera Slika na ekranu dobijena je uz pomoć visoko profesionalnog skenera "Howtek Scanmaster", o kome pišemo u ovom broju

Na osnovu Mišljenja Sekretarijata za informacije SR Srbije broj 413-01-48 od 4 2. 1991 na ovaj časopis plaća se osnovni porez na promet po povlašćenoj tarifi od 3 odsto.



CD-ROM

HITACHI – TREĆA GENERACIJA

Hitachi je izašao na tržište sa dva CD-ROM drajva treće generacije CDR 3600 i CDR 3650. U odnosu na prethodne, novi modeli imaju niz poboljšanja, kao što su kraće vreme pristupa (0.45 s), veći bafer, redizajniranu mehaniku i elektroniku, mehanizam za automatsko čišćenje optike, dvostruka vratašca za disk i mogućnost korišćenja audio CD diskova. Oba modela raspolažu sa dva moda – u prvom kapacitet iznosi 533 MB uz brzinu prenosa od 153.6 KB/s, dok u drugom modu kapacitet raste na 630.7 MB, a brzina prenosa na 175.2 KB/s, uz vreme pristupa iznosi 550 ms. Model CDR 3600 raspolaže baferom od 32 KB, a na računar se priključuje preko bus-interfejs kartice CD-IF35A ili CDR-IF18 A (do osam *daisy-chain* povezanih uređaja), model CDR 3650 je opremljen baferom od 64 KB i standardnim SCSI kontrolerom. Dimenzije uređaja su slične dimenzijama 5.25" flopi drajva – 146 x 41.3 x 203 mm – a težina iznosi približno 1.6 kg.

Kontakt adrese:
Hitachi Sales Europa GmbH, Rungedamm 2, D 2050 Hamburg 80, tel. 40 734 110
Hitachi Sales (U.K.) Ltd., Hitachi House, Station Road, Hayes, Middlesex UB3 4DR, England, tel. 1-848-8787

Portabli

286 NA DLANU

Dok većina današnjih palmtop računara više liče na moderne elektronske notese nego na prave računare, honkonška firma „Palmcom International Ltd." je predstavila model koji bi mogao da bude prvi pravi palmtop računar na svetu. U kućištu veličine 213 x 118 x 30 mm nalazi se prvi 286 procesor dizajniran specijalno za palmtop računare AMD AM286LX 16, koji predstavlja veliki korak napred u odnosu na dosadašnje 8088-5MHz mašine što dominiraju tržištem. Podršku procesoru pruža „Palmcomov" specijalno dizajnirani ASIC sa oznakom PA208LM-A1, koji sadrži kontrolnu logiku za pogon LCD displeja, IC-card drajvova, tastature i napajanja. Druga novost koju ovaj računar donosi je superuvrnuti CGA LCD displej rezolucije 640×200 tačaka bez pozadinskog osvetljenja sa maksimalnim uglom gledanja od 60°. Ovakve osobine mu omogućavaju rad sa mnogim aplikacijama koje ne mogu da rade na većini današnjih palmtop računara.

Jedna od bitnijih mogućnosti je 120-pinski ekspanzioni port na koji se može priključiti opcioni „Super I/O Expansion Interface" ili „Multifunction System Docking Station". I/O interfejs sadrži standardni serijski i printer port, interfejs za eksternu tastaturu i monitor, a multifunkcijski interfejs omogućava priključivanje flopija i hard diska ili standardnih AT Bus ekspanzionih slotova. Na taj način se većina funkcionalnih ograničenja ove klase računara mogu prevazići dodavanjem perifernih uređaja preko ekspanzionog slota. Sistem zaokružuju dva PCMCIA/JEDIA IC-card drajvova smeštenih sa prednje strane ispod tastature – baš kao što su prvi PC-ji koristili dva flopija, jedan za programe a drugi za podatke, tako i ovaj računar koristi dva IC-card drajva. Budući da se na tržištu može naći vrlo ograničen broj programa na IC karticama, napravljen je *IC card download kit* za prenos podataka sa klasičnog PC-ja na IC kartice i obrnuto. *Personal Information Manager* ugrađen u ROM je deo standardne opreme i sadrži rokovnik, adresar, notes i kalkulator.

Kontakt fax (Hong Kong): 852-753-5764

Pomoćni uređaji

NAJKOMPAKTNIJI BAFER NA SVETU

FB 190 je oznaka za eksterni bafer za štampač tajvanske firme „ATEN" veličine 76 x 51 x 20 mm koji raspolaže posebnim metodom za kompresiju podataka sa stepenom kompresije od 15.85%. Bafer raspolaže sa maksimalno 4 MB (64 i 256 KB, 1MB) RAM-a, brzinom prenosa (komprimovanih podataka) od 45 KB/s (20 KB/s za nekomprimovane podatke), *File Copy* opcijom koja dozvoljava pravljenje do 255 kopija i automatski self-test. Uređaj se postavlja u Centronics port štampača i može se kaskadno povezati sa dodatnim jedinicama. Napajanje 9V, 100 mA.



Kontakt adrese:
Aten International Co. Ltd., 12F No. 101 Sung Chiang Rd., Taipei 10428, Taiwan, R.O.C., tel. 886 2 5047270, fax 886 2 5047282
Aten GmbH, Gutenbeergstr 7, D-7024 Filderstadt

ISTO ALI MALO DRUGAČIJE

C-386S je oznaka laptopa baziranog na 386SX 16 procesoru firme „Comax Systems" – na prvi pogled sasvim obična konfiguracija koja uključuje 1 MB RAM-a (proširivo do 5 MB), hard diskom od 40 MB/25 ms, 1.44" flopijem, monohromatskim VGA LCD displejem sa pozadinskim osvetljenjem i 16 nijansi sivog, podrškom LIM/EMS 4.0 i podnožjem za opcioni matematički koprocesor. Računar raspolaže sa jednim paralelnim i dva serijska porta, priključcima za eksterni flopi i monitor, ekspanzionim slotom za jednu poludužinsku ISA karticu i jednim specifičnim slotom. Ono što ovaj računar odvaja od većine drugih je tastatura sa 82 tastera koja se može odvojiti od kućišta i čak zameniti standardnom AT tastaturom. Za napajanje su zaduženi dvostruki NiCd akumulator i ugrađeni ispravljač/punjač.

Kontakt adresa: Comax Systems Ltd., 13/F Unit 2, Wyler Centre Phase II, 200 Tai Lin Pai Rd., Kwai Chung N.T., Hong Kong, tel. 852 481-3015, fax 852-418-0977

WORD PERFECT 5.1 GOVORI RUSKI

Počev od ove godine *WordPerfect 5.1* se proizvodi i distribuira i na ruskom širom sveta. Ceo paket je „lokalizovan", što znači da nije preveden samo program već i sva prateća dokumentacija i priručnici. Naravno, u paket je uključen i ruski rečnik za kontrolu pravopisa koji je razvijen još pre izvesnog vremena.



Tržište

RUSI DOLAZE

Britanska firma „Dialogue UK Ltd." ugovorila je uvoz softvera iz bivšeg Sovjetskog Saveza na zapadno tržište. Trenutno dostupni paketi obuhvataju određen broj programerskih alata, tekst editora, baza podataka, programa za zaštitu od kopiranja i anti-virus programa. Svi programi su napravljeni u vodećim ruskim akademskim institucijama po veoma visokim standardima. Prema rečima predstavnika firme „Dialogue", zbog ograničenih mogućnosti nabavke hardvera u njihovoj zemlji, ruski programeri su naučili da budu vrlo precizni i disciplinovani, tako da su svi programi veoma čisto i efikasno napisani uz minimalne memorijske zahteve, što za korisnike predstavlja veoma dobru kupovinu.

Portabli

TRIO IZ KOREJE

Korejska firma „Cordata", član „Daewoo" korporacije, na nedavno održanom CeBIT-u se predstavila sa tri notebook računara.

CS 2200/SX, najmanji po karakteristikama, opremljen je klasičnim 386SX procesorom koji radi na 20 MHz, 1 MB RAM-a koji se može proširiti do 5 MB na ploči, 2.5" hard diskom od 40 MB, 3.5" flopijem i monohromatskim VGA LCD displejom. Displej je izrađen na bazi teh



nologije dvostrukog filma sa superuvrnutim tečnim kristalom i raspolaže bočnim osvetljanjem i rezolucijom od 640×480 tačaka a firma tvrdi da je u stanju da prikaže 32 nijanse sivog Računar podržava i eksterni monitor do rezolucije od 800×600 tačaka dodatni flopi i eksternu AT tastaturu Opciono se može nabaviti Fax/Modem koji radi na 9600 bps (fax) i 2400 bps (modem) a u standardnu opremu spadaju jedan 9 pinski RS 232 i jedan *Centronics* port koji istovremeno služi i za priključivanje eksternog flopija Računar koristi sesl 1 2 V NiCd bateriju čija trajnost je deklarisana na dva sata normalne upotrebe a kao opcija se nudi i akumulator kapaciteta 2 2 Ah čija trajnost je deklarisana na tri sata neprekidnog rada naravno, u standardnu opremu spada i mrežni adapter Dimenzije su 304×254 x45 mm a težina iznosi 3 1 kg

CS-1000 je opremljen novim „Intelovim" procesorom 386SL koji radi na 25 MHz i koji se opciono može dopuniti odgovarajucim matematičkim koprocesorom Mašina je standardno opremljena sa 2 MB RAM a (proširivo do 8 MB putem dodatne kartice) 32 KB keša 1 44 MB flopijem 60/80 MB hard diskom i monohromatskim VGA LCD displejom dijagonale 10" sa pozadinskim osvetljenjem i rezolucijom od 640×480 tačaka koji podržava 64 nijanse sivog Računar je opremljen po jednim Centronics i RS 232 portom priključcima za eksterni VGA monitor i standardnu tastaturu Tu je i jedan 50-pinski SCSI konektor a kao opcija se nudi interni fax/modem brzina 9600/2400 bps Za napajanje je zadužen jedan NiCd akumulator od 2 8 čiji je vek trajanja deklarisan na četiri sata normalnog rada i jedan mrežni adapter/punjač U cilju ekonomisanja potrošnjom na raspolaganju su tri stepena redukcije – idivi dualni *power-down* mod *Resume & Suspend* mod i puni *Power-down* mod Dimenzije su 221 x 284 x 46 mm težina 3 13 kg a cena iznosi 3 495 USD (uključujući i porez od 15%)

Najatraktivniji deo trojke CS 1500 predstavlja kolor varijantu prethodnog modela Opremljen je kolor VGA LCD displejom dijagonale 9 koji podržava 256 boja pri rezoluciji od 640×480 tačaka i koji je u stanju da radi istovremeno sa eksternim VGA monitorom Izvesne razlike u odnosu na model CS-1000 se primećuju i u količini ukupne memorije koja se može proširiti na 10 MB Pored toga na raspolaganju je i opcioni Nikl-metal-hidridni akumulator dužeg veka trajanja Standardno napajanje dimenzije i težina su isti kao i kod monohromatskog modela uz cenu od 4 925 USD

Kontakt adresa Cordata Technologies Inc Kruisweg 821 2132 NG Hoofddorp The Nederlands tel/fax +31 2503 36530 tlx. 71333 Mrs Catrineke den Daas

Windows

LETRASTUDIO DRAW/PLOT

LetraStudio Draw Windows je naziv paketa čuvene firme „Letraset" namenjen pre svega grafičkim dizajnerima Program omogućava izvođenje niza specifičnih poslova kao što je izrada mapa plakata logoa signatura separacija boja grafički dizajn i izrada kontura za isecanja U tu svrhu je pored standardnih crtačkih funkcija opremljen bogatim setom opcija kao što su određivanje skale u rasponu od 1/1000 mm do 10 km izvlačenje i formiranje pozadinskih linija po kojima teče tekst razne manipulacija sa slikom i objektima (zakrivljenja perspektive senčenja projekcije, ogledala simetrije itd)

Funkcije su podeljene na pet grupa – funkcije za tekst omogućavaju direktnu obradu PostScript tekstualnih fajlova sa svim tipografskim atributima i setovima znakova uključujući i ligature akcente i alternativne znake a tu su i razne distorzije zakrivljenja i lepljenja na zadate oblike Grafičke funkcije služe za definisanja osnovnih objekata (linija, poligona krugova elipsi) kompleksnih kontura premeštanje skaliranje distorziju simetriju umnožavanja i menjanje boja Ulazno-izlazne funkcije obuhvataju unos fajlova u TIFF Adobe i Illustrator formatima kao i izlaz u Adobe Illustrator i EPS (*Encapsulated PostScript*) formatima Tu su i separacija boja direktna podrška sekačima matričnim laserskim i PS štampačima kao i PostScript osvetljivačima (u Letraset Linotip Monotip Adobe i Berthold formatima) Pomoćna funkcije obuhvataju postavljanje pomoćnih linija i rastera određivanje uglova i zumiranje do 64 puta kao i odgovarajuće umanjenje Peta grupa obuhvata sva preostale funkcije koje ne spadaju u neku od prethodnih grupa kao što je funkcija za određivanje koordinata

LetraStudio Draw Windows zahteva minimalno AT 286 sa 2 MB slobodnog RAM a odgovarajućom (bar VGA) grafičkom karticom i monitorom mišem hard diskom sa najmanje 20 MB slobodnog prostora DOS 3 0 ili viši i *Windows 3 0* ili viši Preporučuje se i ugradnja matematičkog koprocesora a za maksimalni kvalitet i izlaz na PostScript štampač ili osvetljivač

LetraStudio Plot nudi dodatne specifične funkcije kao što su digitalizovanje na ekranu digitalizacija kontura obrada Bezierovih linija i segmenata definisanje spoljnih i unutrašnjih kontura (*Inline/Outline*) operacije nad ravnima i površima itd Minimalna konfiguracija je u odnosu na *Draw* verziju bazirana na 386 mašini što većeg takta sa koprocesorom ili (bolje) 486 ici i minimalno 4 MB slobodnog RAM a Podrazumeva se da *Windows* radi u 386 (Enhanced) modu

Kontakt adrese
Letraset Deutschland GmbH D-6000 Frankfurt/Main 60 Mergentahlerstr 6 Deutschland
Letraset Vertriebsges m b H A-1160 Wien Panikengasse 34 Österreich
Letraset (Switzerland) AG CH-8152 Glattbrugg Kanalstraße 15 Switzerland



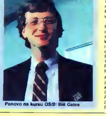
Bil Gejts, direktor „Microsofta":

ODUSTAJEMO OD *WINDOWS-A*

"Microsoft" je „kapitulirao" pred IBM-om u pogledu OS/2 konstatuje se u izjavi Bila Gejtsa (Bill Gates) direktora „Microsofta", objavljenoj u aprilskom broju nemačkog mesečnika „PC Welt"

"Konkretno govoreći OS/2 je bolji *Windows* od *Windows*-a Sve što računarsko tržište traži nalazi se u OS/2 i mi moramo da revidiramo naš sud o njemu Microsoft sada napušta razvoj *Windows*-a i ponovo pristupa IBM-u " ( )

"OS/2 v 2 0 nudi sve U ovom trenutku on sa prihvatljivim performansama funkcioniše jedino na IBM mainframe mašinama Međutim mi ćemo se ponovo vratiti PC-ima "


Ponovo na kursu OS/2: Bill Gates

OCR

NOVI RECOGNITA PAKETI

Recognita Corp.' je izbacila na tržište tri nova OCR paketa – reč je zapravo o dve nove verzije već postojećih programa i jednom sasvim novom paketu. *Recognita Plus 1.2* donosi poboljšanje brzine i višejezičkih sposobnosti uz pročišćavanje postojećih opcija, kao i podršku za *HP AccuPage* tehnologiju koja automatski određuje intenzitet i čisti pozadinu od neželjenih tačkica i uzoraka koji nastaju kao nusprodukt skeniranja, čime se dobija na kontrastu teksta i kvalitetu očitavanja. *Recognita Plus* je napisana specijalno za PC okruženja i intenzivno koristi usluge procesora. Radi pod *Windows 3.0* okruženjem, DOS-om i koristi DOS komandnu liniju i bač fajlove. Verzija 1.2 podržava skoro sve 300 i 400 dpi flatbed skenere koji postoje na tržištu (preko 80 modela).

Recognita Plus 1.2 je u stanju da čita mešani tekst sa veoma malim rastojanjem između slova koja mogu biti i podvučena. Razdvajanje kompleksnih struktura sa tekstom i slikama je unapređeno, a podržani su i komprimovani formati slika kao što su TIFF i PCX. Nova verzija podržava 22 evropska jezika uz prikazivanje na ekranu, a pored toga može da čita bilo koji jezik pisan latiničnim pismom, uključujući i neevropske – ukupno preko 80. **Correct and Learn** opcija omogućava ligature i stringove od maksimalno 10 karaktera, uklanjanje neželjenih karaktera i učenje ligatura od po 2 karaktera. Podržano je nekoliko ASCII izlaznih formata i preko 30 različitih tekst-procesora, unakrsnih tabela, uključujući i *Excel 2.1*, *Lotus 1-1-3* i *Word for Windows*.

Recognita Select 1.0 predstavlja program za rad isključivo sa HP „ScanJet"-om IIP uz podršku *HP AccuPage* tehnologije koja automatski vrši OCR pretprocesiranje i primenjuje specijalne tehnike prepoznavanja na različitim delovima strane. Program obrađuje i gust tekst sa mešanim fontovima i podvlačenjima, razdvaja kompleksne forme teksta i grafike i podržava nekoliko komprimovanih formata slika. Za rad je potrebno imati *Windows 3.0* i bar 1 MB slobodne memorije.

GO-CR 2.0 je OCR program prilagođen ručnim skenerima. Nova verzija, pored poboljšanja starih mogućnosti, raspolaže i mogućnošću interaktivne obrade teksta – tokom upotrebe tekst-procesora. GO-CR se startuje iz menija i posle prepoznavanja automatski unosi tekst direktno u tekst procesor. Za rad je potreban PC sa *Windows 3.0* okruženjem i neki od podržanih ručnih skenera (na primer „E-Tech", „Genius", „Logitech", „Marstek", itd). GO-CR prepoznaje neproporcionalni, proporcionalni ili kucani tekst u više kolone ili u obliku čistog dokumenta, uz eliminisanje grafike. Podržano je preko 80 jezika baziranih na latiničnom pismu, a 12 jezika ima i ekransku podršku.

SZKI Recognita Corp., H 1015 Budapest I Donati u 35-45 Hungary, tel. 361 201 8925, 361 201-8452, 361 201 8535, fax 361 201 7607, 361 201-7773

Štampači

POSTSCRIPT SLAJDOVI U BOJI

SpectraStar 450" je za sada jedini termalni kolor printer koji omogućava štampanje PostScript slajdova od 35 mm. To se postiže uz pomoć posebnog slajd rikordera, kojim se svaki „SpectraStar" kolor printer može dograditi do modela 450 uz cenu od oko $6.800. Sistem je



zasnovan na RISC procesoru koji omogućava pogon dva izlazna uređaja, i to je prvi put da se jedan sistem može koristiti za štampanje i na papiru i na filmu. Slike se mogu štampati i sa hard diska od 32 MB ugrađenog u printer, čime se kompjuter oslobađa za druge poslove. Komplet koji se sastoji od „SpectraStar 450" printera sa 80960 RISC procesorom koji radi na 20 MHz, 32 MB hard diskom za baferovanje ulaza i smeštanje fontova i slika, hardverskom PostScript kompatibilnošću i 6 MB RAM-a i „SlideMaker" rikordera se prodaje po ceni od $19.760 i predstavlja najjači model „SpectraStar" serije. Tzv. *low-end* model je „SpectraStar 410", koji se prodaje po ceni od $5.900. Svi „SpectraStar" modeli podržavaju i PC i „Macintosh" platforme.

Kontakt adresa: General Parametrics Corporation, 1250 Ninth Street, Berkley, CA 94710, tel. 510 524 2950, 800-223-0999, fax 510-524 9954

Procesori

OVA NOVA KONTROLERA

Čuvena firma „VLSI Technology" je započela proizvodnju dva nova kontrolerska čipa sa oznakama VL82C486 i VL82C425. Prvi predstavlja kompletan kontroler na jednom čipu za 486SX i 486DX sisteme koji rade na frekvencijama do 33 MHz. Reč je o visoko integrisanom CMOS integrisanom kolu koje zamenjuje devet periferijskih jedinica na matičnoj ploči: dva DMA kontrolera, programabilni interval-tajmer, takt-generator, dva interapt kontrolera, memorijski maper i bus kontroler. Pored toga, novi čip sadrži kontrolu paritela i logiku za generisanje pariteta, kontrolu *burst* moda i još niz mogućnosti. Integracijom logike za kontrolu memorije VL82C486 omogućava direktnu podršku do 64 MB sistemskog DRAM-a, raspoređenog u četiri banke sa maksimalno 4M DRAM čipovima bez eksternog baferovanja. Pored toga, novi kontroler omogućava postizanje značajno veće brzine bus-a, a takođe integrisana podrška za pariterje lokalnog bus-a omogućava maksimizaciju performansi periferija kao što je VGA kontroler upotrebom brzog CPU bus-a. Čip se pakuje u PQFP (*Plastic Quad Flat Pack*) kućište sa 208 nožica.

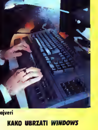
VL82C425 predstavlja integrisani kontroler drugostepenog keša za 486 mašine na brzinama do 33 MHz. Čip sadrži komparator, linijski bafer i kontrolnu *write-back* logiku i podržava direktno mapiranje sa eksternim tagovima i upotrebu regularnih SRAM čipova za keširanje tagova i podataka. Mogu se koristiti keš-setovi od 64, 128, 256 i 512 KB i 1 MB sa SRAM čipovima od 25 do 20 ns za podatke (33-40 MHz) i 20 do 15 ns za tagove (33-40 MHz). Čip podržava dva režima rada – 2-1-1-1 *Burst Mode* sa 2 banke SRAM-ova za podatke ili 2-2-2-2 *Burst Mode* sa jednom bankom. Novi čip koristi 1 mikronsku CMOS tehnologiju i pakovan je u 128-pinsko MQFP (*Metric Quad Flat Pack*) kućište.

Kontakt adresa: VLSI Technology Inc., 8375 South River Parkway, Tempe, Arizona 85284, tel. 602-752-8574, fax 602 752-6000

Tastature

BRAJOVA TASTATURA

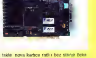
Na slici je prikazano jedno rešenje koje omogućava slepima da koriste računar. Reč je o kombinaciji standardne tastature sa tastaturom za Brajovo pismo koja je smeštena ispod standardnog dela.



Drajveri

KAKO UBRZATI *WINDOWS*

Jedno od mogućih rešenja za ubrzanje *Windows*-a je grafička kartica *GraphicsEngine 70* firme „Actix", projektovana specijalno za ubrzavanje *Windows* okoline. Kartica je predstavljena na CeBIT-u i prema do sada objavljenim testovima, znatno je brža od ranije verzija koja je radila na 55 MHz. Nova kartica radi na 70 MHz i značajno ubrzava sve video operacije – pojedine operacije i do čitavih 15 puta u odnosu na standardne video kartice sa Tseng Labs ET-4000 čipom. Pored impresivnog rad-



nog takta, nova kartica radi i bez stanja čekanja uz asistenciju posebnog čipa za ubrzanje grafike. Ubrzanje najviše dolazi do izražaja u CAD aplikacijama, host interfejsima, grafičkom oblikovanju i obradi teksta, a cena koju za to treba platiti iznosi između 600 i 700 DEM.

Kontakt adresa: Actix Systems Inc., 4633 Old Ironsides Drive, Suite 310, Santa Clara, California 95054, tel. 408 986 1625, fax 408 986 1646

LAN NA MLAZNI POGON

Iz IBM-a stižu zadivljujuće vesti za korisnike lokalnih računarskih mreža. Ovaj svetski gigant u oblasti računarske tehnologije priprema nešto stvarno vredno pažnje – novu *Token Ring* mrežu koja će prenositi podatke brzinom 100 megabita u sekundi.

Dragan Pleskonjić

U oblasti lokalnih mreža primat je dugo držao „Xeroxov" *Ethernet* sa brzinom prenosa podataka od 10 Mbita/s po tzv. debelom kablu (*thick cable*). *Ethernet* mreže inače koriste pristupni protokol CSMA/CD prema standardu IEEE 802.3. Mnogi proizvođači su u međuvremenu plasirali na tržište svoje sopstvene mreže, ali one dugo nisu ugrožavale *Ethernet*. Nakon nekog vremena IBM je izbacio na tržište svoju *Token Ring* mrežu sa 16 Mbita/s (opciono 4 Mbita/s) o kojoj je pisano u „Računarima" 80. Ova mreža je prekinula dominaciju *Ethernet* mreže u pogledu brzine. Trenutna situacija u ovom području izgleda otprilike onako kako je to prikazano u tabeli uz ovaj tekst. Tabela, uz izvesna ažuriranja, potiče iz knjige „Computer Networks, Protocols, Standards and Interfaces" (autor Uyless Black).

Očito povođen nezajažljivim apetitima i oštrim zahtevima korisnika mreža, IBM trenutno intenzivno radi na pripremi novog komercijalnog proizvoda – *Token Ring* mreža na 100 Mbita/s. IBM je još ranije odabrao *Token Ring* mrežu za svoj strateški proizvod. Uz adekvatnu brzinu, ova mreža omogućuje povezivanje različitih tipova računara – PC, PS/2, AS/400, RISC/6000, veliki IBM i drugi računari (*mainframe*) – kao i povezivanje sa drugim mrežama, lokalnim i rasprostranjenim (LAN i WAN).

OSLOBAĐANJE ZNAKA

IBM je za svoj *Token Ring* primenio nešto izmenjeni IEEE 802.5 standard, a sada za novi *Token Ring* od 100 Mbita/s koristi još jednu modifikaciju, odnosno unapređenje poznato pod nazivom *Early Token Release* (ETR) ili u doslovnom prevodu „ranije oslobađanje tokena (znaka)". Radi se o tome da se kod klasičnog IEEE 802.5 protokola token oslobađa tek nakon što podaci koji su poslani od izvorišnog čvora kroz mrežu prođu pored odredišne stanice, budu u njoj kopirani (copy bit postavljen na „1") i opet stignu do izvorišne stanice. Izvorišna stanica „skida" poruku sa prstena, to jest oslobađa kanal i predaje token sledećoj stanici. To je približan način na koji se stvari odvijaju. Izvorišni čvor, nakon što je poslao podatkovni okvir (*data frame*), šalje prazne (*idle*) znakove dok okvir ponovo ne stigne do njega. Ovim se izaziva izvesne latencija, to jest skrivene besposlenosti prenosnog kanala. Tipična latencija za 4 Mbita/s *Token Ring* je 50-100 *idle* znakova, dok za 16 Mbita/s latencija može dostići i preko 400 *idle* bajtova.

U varijanti *Token Ring*-a sa ETR, token se oslobađa, odnosno dodaje sledećoj stanici odmah nakon što se pošalje podatkovni okvir. Time se izbegava stanje besposlenosti kanala, to jest na kanalu se ne pojavljuju dugi repovi praznih znakova nego samo tokeni i podatkovni okviri. Ovim se znatno povećava efikasnost prenosa, to jest brzina prenosa informacionih bitova. Ovo unapređenje protokola izgleda kao jednostavna ideja, ali je realizacija prilično složena.

Drugi elementi koji utiču na postizanje ovako velike brzine verovatno će biti odgovarajući kabling sistem i specijalni mrežni adapteri. Mrežni adapteri će imati znatno više RAM-a od dosadašnjih, a možda čak i svoj posebni procesor. To ostaje da se vidi. Treba očekivati da će ova mreža zadovoljiti i najoštrije zahteve korisnika, a o ceni komponenata će se tek čuti.

Prioritetno područje primene ove superbrze mreže moglo bi biti međusobno povezivanje LAN mreža i povezivanje LAN mreža na velike (*mainframe*) računare (*backbone* mreže). Sigurno je da će biti i korisnika koji će i u okviru svojih pojedinačnih lokalnih mreža, ako su im performanse kritične, koristiti 100 Mbitni/s Token Ring. Uostalom, zahtevi velikih korisnika su i naterali proizvođače da krenu u jedan ovakav razvoj.

Ranije je brzina 100 Mbita/s bila moguća korišćenjem isključivo optičkih kablova. Ovakve mreže su definisane ANSI standardom X3T9.5 pod nazivom *Fiber Distributed Data Interface* (FDDI). Cena ovakve mreže je, naravno, znatno veća od cena mreža zasnovanih na metalnim provodnicima, ali su joj i performanse znatno bolje. IBM će svojim novim proizvodom verovatno postići dosta nižu cenu u odnosu na mreže sa optičkim kablovima.

IBM *Token Ring* na 100 Mbita/s koristi sličnu prioritetnu shemu kao i FDDI. Svaki uređaj na prstenu može biti sinhroni ili asinhroni. Veliki računari (*mainframe*) priključeni na *Token Ring* imaće veći prioritet, a takođe i serveri u mreži u odnosu na druge stanice u mreži.

BUJICA PAKETA

U skladu sa težnjama da se stvore što brže mreže, „Novell", kao jedan od vodećih proizvođača u ovoj oblasti, radi na proširenjima i prilagođenjima svog *Netware* mrežnog operativnog sisteme. Poznato je proširenje pod pomalo šaljivim nazivom „paket burst" (navala, juriš paketa).

"Packet-burst" predstavlja tehniku koja je dodata na poznate *Kermit*, *XMODEM* protokole, čime su formirane osnove *ZMODEM* protokola. Time se omogućava slanje podataka između File Servera i radne stanice u dugom nizu paketa, bez potrebe da se da poseban zahtev za svaku malu sekciju datoteke. „Novell" preuzima i oplemenjuje ovaj protokol. Omogućeno je grupno potvrđivanje paketa, pri čemu se jednim paketom potvrđuje prijem čitavog niza prethodno primljenih paketa. Na taj način se izbegava čitav niz potvrda pojedinačnih paketa i štedi vreme prelaska iz stanja predaje u stanje prijema. Pri klasičnom načinu pojedinačnog potvrđivanja može u međuvremenu doći i do preuzimanja kanala od strane drugih stanica, što je izvesni gubitak vremena. U slučaju primene *packet burst*-a, prenos velike datoteke se obavlja „u letu" nizom paketa i sa znatno manjim brojem paketa potvrde. Drugi termin za *packet-burst* je *sliding windows*.

SARADNJA VELIKIH

Očigledno je da IBM i „Novell" sve više usklađuju svoje poteze. Cilj je izgradnja mreža sa što boljim performansama i učvršivanje pozicija na ovom značajnom segmentu računarskog tržišta. IBM osnažuje svoj hardver, dok „Novell" ide napred sa softverskom podrškom koja takav hardver maksimalno koristi. Već su potpisani sporazumi između ove dve firme, što samo ide u prilog sve brojnijim korisnicima računarskih mreža širom sveta.

Napomena: Neki od podataka navedeni u ovom tekstu potiču iz časopisa „Byte", januar 1992. godine.

Tabela 1 IBM PC mreže

Proizvođeč	Naziv	Kapacitet	Medij	SignalProtokol
3COM	Ethershere	10.0	CO	BAcsma/cd
AST Research	PCnet II	0.8	TW	BAcsma/cd
AT&T	StarLAN	1.0	TW	BAcsma/cd
Corvus	Omninet	1.0	TW	BAcsma/cd
Datapoint	ARCNET	2.5	CO	BAtoken
Davong	Multilink	2.5	CO	BAtoken
Digital Micro	HiNet	0.5	TW	BApolling
Gateway	G-Net	1.4	CO	BAcsma/cd
IBM	PC Network	2.0	CO	BRcsma/cd
IBM	Token-Ring	16.0	4.0	TW FIBA token
Intel	StarLAN	1.0	TW	BAcsma/cd
Nestar	PLAN 2000	2.5	CO	BAtoken
Novell	Netware/S	0.6	TW	BApolling
Orchid Tech	PCNET	1.0	CO	BAcsma/cd
PC Office	Net Board	2.5	TW	BAcsma/cd
PROLAN Corp	PROLAN	2.5	CO	BAtoken
Proteon	ProNET	10.0	CO TW IN	BAtoken
Quadram	Quadnet VI	1.4	CO	BAcsma/cd
Quadram	Quadnet IX	10.0	TW	BAtoken
Santa Clara	PC-Net	1.0	TW	BAcsma/cd
Standard Micro	ARCNET	2.5	CO	BAtoken
Sytek	Localnet20	2.0	CO	BRcsma/cd
Ungerman-Bass	Net/One	5.0	CO	BRcsma/cd
Ungerman-Bass	Net/One	10.0	CO	BAcsma/cd
Wang	Wangnet	2.5	CO	BRtoken
Xerox	Ethernet	10.0	CO	BAcsma/cd

Medij: **CO** koaksijalni kabl, **FI** optički kabl, **IN** infracrvena veza, **RS** RS-232, **TW** žičana parica
Signal: **BA** Baseband, **BR** Broadband

Fiber Distributed Data Interface (FDDI) koristi optička vlakna i omogućuje brzinu 100 Mb/s

IBM Token Ring sa Early Token Release (ETR) proširenjem treba da omogući brzinu 100 Mb/s

U POTRAZI ZA IZGUBLJENIM USPEHOM

Poslednja inkarnacija „operativnog sistema 90-tih" najzad ispunjava neka obećanja. Ali još uvek je dosta želja koje tek treba ispuniti.

Obzirom da je „Microsoft" postigao veliki uspeh u prodaji *Windows*-a *3.0* promenio je svoje planove vezane za novu generaciju operativnih sistema. U „Microsoftu" su objavili planove („Računari" su već pisali o tome) da naprave sopstveni *New Technology* (*NT*) operativni sistem. IBM je na neki način ostavljen na cedilu obzirom da je radio zajedno sa „Microsoftom" na razvoju nove verzije OS/2. No to nije pokolebalo ljude iz IBM-a, jer su sami napravili novu verziju OS/2 sa mogućnostima koje Microsoft nikada nije podržavao. Tu je uključena potpuna podrška DOS-a (uključujući *device driver*-e), rad u zaštićenom režimu za DOS i *Windows* aplikacije, 32-bitna grafička mašina u obliku *Presentation Manager*-a (PM) i potpuno novi grafički izgled.

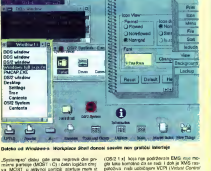
INTEGRISANA PLATFORMA

IBM je nazvao novi sistem „integrisanom platformom" i on to zapravo i jeste. Računari na koje smo instalirali verziju 2.0 pokreću vrlo različite vrste softvera: 16-bitne OS/2 aplikacije, DOS aplikacije koje rade u stvarnom i zaštićenom modu, kao i *Windows* programe. Četvrta i najprimamljivija vrsta softvera – 32-bitni OS/2 programi čekaju na krilima. Veliki broj proizvođača OS/2 softvera već pravi verzije za 2.0, a IBM je prikazao nekoliko njihovih prvih varijanti. Trenutno još uvek nema na raspolaganju puno 32-bitnih PM aplikacija. Delimični krivac za to je i „Microsoft", jer njegov po zlu čuveni *Software Development Kit*, čija je cena oko 2600 USD, nikada nije radio kako treba. Sada IBM plasira sopstveni kompajler za 2.0, a velike i poznate firme kao što su „Borland", „Digitalk", „Jensen & Partners International" i „Watcom" planiraju da ponude alternativne programerske alate.

Pisci programa za OS/2 su tokom prošle godine postali, često i protiv svoje volje, *Windows* programeri. Sa programima nezavisnih proizvođača i „Microsoftovim" 32-bitnim *Windows NT*, koji se očekuje sledeće godine, OS/2 nije lako privlačan za veliku i jedinstvenu podršku njegovom 32-bitnom PM-u. Međutim IBM je izbacio verziju 2.0 i obećao da će to biti „bolji DOS od DOS-a, bolji *Windows* od *Windows*-a". Po našem mišljenju OS/2 2.0 bi mogao biti bolji DOS u mnogim namenama i bolji *Windows* dostojan poštovanja.

Čak i prvi zaljubljenici i korisnici OS/2 moraju povremeno da koriste stari DOS. Jedna od najlepših mogućnosti novog OS/2 je da na jasan i elegantan način rešava taj problem. Kada instalirate sistem, možete upotrebom FDISK-a da kreirate specijalnu particiju od 1MB u kojoj će biti MOST (*Multiple Operating System Tool*). Ostatak diska može biti podeljen na još dve primarne particije (od kojih je jedna C) i dodatne particije, koje će sadržati logičke diskove (D i dalje). MOST može odabrati operativni sistem sa jedne od primarnih particija ili bilo kojeg od logičkih diskova.

Konkretno, stvar smo probali na 800 Mb



Daleko od *Windows*-a *Workplace Shell* donosi sasvim nov grafički interfejs

„Systempro" disku gde smo napravili dve primarne particije (MOST i C) i četiri logička drajva. MOST u aktivnoj particiji startuje meni iz koga možete odabrati DOS sa diska C ili neku od različitih beta verzija OS/2 ver. 2.0 sa kojima raspolažemo sa logičkih diskova. Mogućnost 2.0 da se pokrene sa logičkih diskova ukomponovana sa MOST i nadograđenim FDISK znatno olakšava pokretanje različitih operativnih sistema, bez potrebe za resetovanjem računara.

POBOLJŠANJA I KOMPROMISI

Kao i *Windows*-i, OS/2 2.0 podržava rad sa prozorima i DOS-om. Možete preći iz jednog moda u drugi preko **Settings** menija, s tim da zgodna kombinacija Alt-Enter iz *Windows*-a neće raditi. Na žalost, ne možete preći iz *Windows* aplikacije u puni OS/2 režim rada. **Cut** i **Paste** u *Windows* aplikacijama funkcioniše na isti način kao u *Windows*-ima. Lepo poboljšanje je mogućnost promene fontova i u DOS i u OS/2 prozoru. Nikada nismo voleli DOS prozor u produženom EGA stilu kako ga je *Windows* prikazivao. Sada nam se mnogo više dopada jedan od stilova koje nudi OS/2.

Kao što je obećano OS/2 se isporučuje sa EMS, XMS (*Extended Memory Specification*) i DPMI (DOS Protected Mode Interface) za rad sa DOS-om. Pokretali smo *FoxPro 2.0* i on je radio dobro, isto kao i u „čistom" DOS-u. To je za nas velika novost jer u prethodnoj verziji (OS/2 1.x), koja nije podržavala EMS, nije moglo tako komotno da se radi. I dok je XMS raspoloživa, naši uobičajeni VCPI (*Virtual Control Interface*) orijentisani programi, kao što su proširena verzija *FoxPro*-a i „Borlandovog" C++, nisu mogli da je koriste. Zašto? OS/2 2.0 odbija da podrži VCPI. Da bi to uradio moralo bi da se napravi više kompromisa sa integritetom sistema.

Na sreću, tu je alternativa DPMI. Napisali smo pomoću „Borlandovog" C++ i „Phar Lapovog" 286/DOS Extendera program za odvajanje jednog megabajta memorije, koji je radio u DOS-ovom zaštićenom modu, a koji je koristio DPMI iz OS/2 2.0. Danas nema mnogo DOS programa koji podržavaju DPMI. Jedan od primera je *Lotus 1-2-3* verzija *3.0*, a ubrzo će ih biti mnogo više, jer postoje programski alati koji podržavaju DPMI (bez toga nema *Windows* kompatibilnosti). Pitanje je koliko konvencionalne memorije zahteva DOS boks? Zavisi. Prvo smo napravili razliku između internih virtuelnih DOS mašina (VDM) i eksternih. Interna VDM, što je osnovni DOS boks, vidi oko 633 K, zato što podržani kod radi u zaštićenom modu izvan DOS-ovog adresnog prostora. Eksterna VDM, koja pokreće normalno DOS jezgro sa DOS particije ili DOS butabilne diskete, vidi manje RAM-a, oko 569 K. U ovakvoj situaciji trebalo bi da možete da „podignete" DOS, ali to još uvek ne radi.

Obe VDM mogu učitati prave drajvere za DOS uređaje i to smo uspešno radili. One nu-

de podršku gornjem memorijskom bloku tako da tamo možete smestiti TSR programe, mada smo mi to uspeli samo sa internim VDM. Bili smo u mogućnosti da pokrenemo *NetWare* i komuniciramo sa CD ROM-om samo na način eksternih VDM.

To što je tako nešto uopšte moguće je, na ravno, ogromna pobeda. Pristup mreži i CD-ROM-u kroz obični DOS mehanizam je velika sreća za OS/2. Postoje i OS/2 drajveri, jer smo pokretali OS/2 1.x mrežu, displej i drajvere za diskove pod verzijom 2.0. Kad god je moguće više volimo da koristimo podršku uređajima kroz DOS boks. Zahvaljujući alfa verziji „Novelovog" OS/2 *NetWare* programa, svi naši programi za OS/2 i DOS su prošli kroz mrežu. Do duše, teško smo dolazili do OS/2 drajvera, dok ih za DOS ima za sve živo pod ovim suncem. *Winodws*-i građeni na DOS-u, su nasledili tu kritičnu softversku osnovu. Sada to OS/2 takođe može, ali pomoću trikova. Lutajući DOS drajver može dovesti *Windows*-e do pucanja, OS/2 mirno i spretno hvata greške bez problema. OS/2 2.0 izvršava DOS programe dovoljno dobro da smo mogli da pokrenemo dve VDM priključene na *NetWare* a na istoj „Tiara" kartici. No, to treba shvatiti samo kao probu, jer ono što je radilo sa „Tierom" i „Systempro" diskom, nije radilo sa „Western Digital" i NE 2000 karticom u „Swan" kompjuteru. Ohrabreni smo, ali ne možemo predskazati kolika će stvarna podrška OS/2 biti za DOS drajvere.

INSTALACIJA I STARTOVANJE

Instalacija OS/2 zahteva upotrebu velikog broja AT disketa (više od jedne kutije), i popul *Windows Setup*-a, odmah vas uvodi u grafički režim rada. Ako pogledate listu sa nazivima datoteka, onda će vam biti jasno zašto je potreban toliki broj disketa. Zajedno sa datotekama koje čine OS/2, tu su i druge kao COMMAND.COM, HIMEM.SYS, PROGMAN.EXE i SYSTEM.INI. To nije *već viđeno* vi stvarno instalirate DOS, *Windows* i OS/2, sve zajedno i odjednom.

I dok su DOS komponente modifikovane za VDM okruženje OS/2 2.0, delovi *Winodws*-a (osim par izuzetaka) su isti kao u originalnom paketu *Windows 3.0*. OS/2 pokreće *Winodws* kao VDM, a *Windows*-i pokreću aplikacije u standardnom modu. Kako? *Windows* aplikacije su DPMI klijenti, a *Windows*-i im (kao DPMI server) dozvoljavaju normalni pristup *extended* memoriji. OS/2 je takođe DPMI server, a IBM je štrpnuo od *Winodws*-a da bi ih napravio klijentom DPMI OS/2 i, istovremeno, serverom za *Windows* aplikacije.

Kada startujete *Windows*-e, ulazite u prave *Windows*-e koje pokreće PM. Iz OS/2 perspektive, prebacivanje iz i u *Windows*-e vršite isto kao i iz DOS-a ili OS/2. Nemodifikovane *Windows* aplikacije rade (i u multitaskingu) sasvim normalno. Jedino što tu nije raspoloživ DOS boks, ali, nema problema, jer OS/2 barata DOS multitaskingom mnogo bolje od *Windows*-a.

Udešeno je da *clipboard* komunicira sa PM-om. Tako smo bili u mogućnosti da kopiramo ikonu iz *Visual Basic*-ovog editora za ikone u *Winodws clipboard*, da se prebacimo u OS/2 i ubacimo je (paste), iz PM *clipboard*-a, u novi OS/2 editor za ikone. Još uvek ne radi *Dynamic Data Exchange* (DDE) server, koji bi trebalo da obezbedi razmenu DDE poruka između *Windows*-a i OS/2. Obzirom da *Windows*-i imaju svoj sopstveni ekran, u *Windows*-ima možete pokrenuti vaš displej drajver za visoku rezoluciju čak i ako nemate ekvivalent za PM.

Windows-e ne možete pokrenuti u proširenom (*enhanced*) modu. OS/2 sadrži dve ključne mogućnosti iz *Windows enhanced* moda: podrška virtualne memorije i DOS multitasking. Neke *Windows* aplikacije zahtevaju rad u proširenom modu. To su one koje koriste 32-bitnu flat-model memoriju. Međutim najvažniji *Winodws* programi koji se proizvode, a mi smo ih dosle probali, odlično rade u OS/2. Šta je sa *Windows*-om 3.1? U IBM-u kažu da će ga ubaciti u OS/2 čim ga „Microsoft" završi.

BOLJI SISTEM

Bolji od *Winodws*-a? Pa, puno ljudi koristi *Windows*-e kao DOS multitasker, a u tom pogledu je potpuno jasno da je OS/2 2.0 bolji od *Windows*-a. Čak i kada su u *Windows*-u 3.1 uklonjeni problemi vezani za „unrecoverable application error", bili smo u mogućnosti da „srušimo" program ubacivanjem slučajnih podataka u nezaštićenu DOS memoriju, koja sve to održava u životu. VDM iz OS/2 2.0 ne poznaje taj problem. Ako poredite *Windows* aplikacije za DOS sa istim za VDM, stvari se dalje komplikuju. U principu, VDM ne radi brže od DOS a na istom hardveru. Nasuprot tome OS/2 2.0 ima brži FAT fajl sistem. Upravljač memorijom kod *Windows*-a vidi DPMI kao fizički RAM. To funkcioniše dobro na našem 12 MB „Systempro" računaru, ali slabo na 4 megabajtnom „Swanu". IBM još uvek puno radi na tome da približi upravljanje memorijom kod *Windows*-a u odnosu na OS/2. To će sigurno zahtevati licenciranje više *Windows* koda od „Microsofta", ali će dramatično unaprediti stvar do vremena kada se 2.0 pojavi na tržištu.

Verzija *Workplace Shell*-a koju smo dobili (vidi sliku) je veoma mnogo napredovala. Liči više na „Mac" nego na *Winodws*, stari PM (koji je i sada prisutan) ili na „Unix" GUI. Za razliku od „Maca", kroz objektnu hijerarhiju ne daje pregled datoteka, već logički pregled referenci datoteka sa podacima, programima, mestu štampanja i dokumentima. Svaki objekt se može otvoriti u nekoliko prozora. Možete otvoriti ikonu-konce da vidite sadržaj ili stranu sa podešavanjima da promenite fontove, boje ili nešto drugo.

Spiralno ukoričena knjižica koja se vidi na ekranu, pored toga što lepo izgleda, služi i za dijalog bokseve i za sistemsku dokumentaciju. *Window List* prati sve poslove u aktivnim prozorima. IBM je na javnoj demonstraciji prikazao još neke mogućnosti (pokretanje programa, štampanje datoteka) koje mi nismo mogli da ispitamo, jer nemamo finalnu verziju.

Obzirom da ovo pišemo pre roka za koji je IBM najavio finalnu verziju, sa strahom očekujemo šta će biti urađeno i šta još treba da se uradi. Može li se postići istovremeno 32-bitni prepis PM-ove grafičke mašine, bolje korišćenje memorije u *Windows*-ima, *Workplace Shell* i dobra integracija DOS, *Windows* i OS/2 delova sistema? Nadamo se da će se te želje ispuniti.

Može li doći do neuspeha OS/2? Teško. OS/2 je uvek bio namenjen individualcima i kompanijama koji cene napredan i dobar operativni sistem. Na nesreću, moraćemo se oslobodili balasta zvanog DOS/Windows da bismo se dohvatili prednosti koje nudi OS/2. Verzija 2.0 obećava da učini ono što je OS/2 trebalo da učini na startu: da objedini, u manjoj meri, familiju PC operativnih sistema i da obezbedi uzlaznu liniju njihovog razvoja. Na kraju, OS/2 nije samo bolji DOS ili *Windows* kako smo želeli, već je to bolji sistem. Mislimo da je to ono što će OS/2 2.0 biti.

Izvor: *BYTE*, autor *Jon Udell* (judell na BIX-u)
Pripremio: *Vladimir Stamenović* (svlada na Sezamu)

ŠTA IMA NOVO

Tržište

FRONT PROTIV PIRATA

Kalifornijska policija je u nedavno izvedenoj raciji otkrila i konfiskovala 3.000 falsifikovanih kopija „Microsoftovog" MS DOS a 5.0 i uhapsila dva funkcionera kompanije „Entech Computer" zbog kršenja zakona o autorskim pravima. Predsednik „Entecha" Či Ming Robert Liu i direktor prodaje Karl Rihard Šihl će biti optuženi i, ako im se krivica dokaže, dobiće kazne zatvora do jedne godine, uz globu za kompaniju od 100.000 USD. Ovaj slučaj je zanimljiv po tome što je to prvi put da „Microsoft" poteže krivični zakon i traži hapšenje i suđenje falsifikatorima softvera u SAD.

Policija je pretresla skladište „Entecha" u momentu kada je planirana isporuka falsifikovanih paketa kupcima, za koje se ispostavilo da su bili jedan prerušeni policajac i jedan „Microsoftov" istražitelj. Zaplenjeni paketi, imitacije softvera vrednog 200.000 USD po uličnim cenama, bili su ilegalne kopije BEC-ovog MS DOS a. „BEC Computer GmbH" je nemačka kompanija koja je od „Microsofta" dobila licencu za prodaju MS-DOS-a pod svojim imenom uz svoje kompjuterske sisteme.

Cele afere se odigravale u maniru krimića: koristeći obaveštenja potkazivača, „Microsoftovi" detektivi su došli do uzoraka ilegalnog paketa sa BEC-ovom etiketom za koji su BEC-ovi stručnjaci utvrdili da je falsifikat. Ono što ih je navelo na sumnju bila je sumnjivo niska cena i „Entechova" spremnost na prodaju pakete nezavisno od hardvera, što se kosi sa standardnim sporazumom sklopljenih između „Microsofta" i firmi kojima je dao licencu. MS DOS 5 Upgrade je jedina verzija MS-DOS-a koja se prodaje nezavisno od hardvera, svaki *standalone* paket je potencijalno sumnjiv kao falsifikat.

Prema Udruženju proizvođača poslovnog softvera (*Business Software Alliance*) softversko piratstvo je do sada nanelo štetu od oko 12-14 milijardi dolara. MS-DOS, koji je u upotrebi na preko 100 miliona kompjutera širom sveta, najpopularniji je operativni sistem za mikroračunare i kao takav je i najviše piratovan. Da bi se legalne kopije što bolje razlikovale od falsifikovanih, „Microsoft" je prošle godine počeo da koristi hologramske etikete koje stavlja na svoja pakete.

Windows

WIN32 API

"Microsoft" je najavio *Win32 API*, programski interfejs koji će omogućiti proizvođačima softvera da prave 32-bitne aplikacije koje će moći da rade i pod *Windows 3.1* i *Windows NT* operativnim sistemima. ***Win32*** tehnologija će omogućiti programerima koji rade u „Microsoft" C-u i C++ da istovremeno prave aplikacije namenjene i jednom i drugom okruženju, a korisnici će dobiti brzinu inherentnu 32-bitnoj arhitekturi aplikacija pisanih za *Win32*.

Win32 tehnologija će biti dostupna kvalifikovanim C i C++ proizvođačima softvera na „Microsoftovoj" Konferenciji profesionalnih programera (*Professional Developers Conference*) koja će se održati od 6. do 8. jula ove godine.

Kontakt telefon (SAD): 001-206-882-8080

ELEKTRONSKA FOTOGRAFIJA

Prijem pravog televizijskog i video programa iz bilo kog kraja sveta preko satelita je do pre samo deceniju-dve predstavljao naučnu fantastiku, a danas je sasvim obična stvar. Dobri, stari fotoaparati i odgovarajuća fotohemijska tehnologija, koji vuku korene još iz prošlog veka, međutim, i dalje suvereno vladaju svojim delom tržišta. Razlog je više tržišne nego tehnološke prirode, ali stvari polako počinju i ovde da se menjaju.



Elektronski fotoaparat nije potpuna novost – velike kompanije kao što su „Nikon" i „Sony" već godinama dodaju svojim kamerama sve veće doze elektronike i inteligencije. Najveću prepreku u tom procesu su predstavljali sistemi za obradu i smeštanje slika na diskove, koji su bili tako veliki da su morali da se nose o ramenu. Razvoj elektronike i digitalne tehnike je omogućio kompletno elektronsko snimanje i smeštanje slika i time potpuno eliminisao potrebu čak i za pomoćnim klasičnim fotografskim procesima. Nova tehnologija je tako postala vrlo privlačna za organizacije kojima je potrebno brzo snimanje i prenos slika – momentalna satelitska veza nema mnogo smisla ako pre slanja fotograf mora prvo da razvije film i zatim izrađuje slike, koje zatim idu na skener da bi se prevele u elektronsku formu.

Oko video tehnologije se izgradio čitav sistem televizijskih vesti, čija brzina emitovanja se već meri skoro isključivo brzinom ljudske reakcije. Takav razvoj stvari je doveo do jedne veoma neobične situacije: danas je lakše i jeftinije preneti „živu" sliku (naravno u boji) preko pola sveta nego snimiti i poslati klasične crno-bele fotografije. Novine i časopisi mogu veoma lako i pouzdano da štampaju slike u boji, ali često moraju da se oslanjaju upravo na zamrznute video slike dobijene preko satelita, budući da klasične fotografije stižu previše sporo.

DA LI JE BRZO...

Prva potpuno elektronska fotografska kamera koja je ušla u masovnu upotrebu širom sveta je bila „Sony"-jeva *Mavica*, lansirana pre skoro punih deset godina. Dimenzija i težine približnih klasičnim foto kamerama srednjeg formata, *Mavica* koristi standardne fotografske objektive koji umesto na film projektuju sliku na CCD (*Charge Coupled Device*). Elektronski signal sa CCD-a se dalje obrađuje pomoću standardne video tehnologije i smešta na interni 2" flopi. Budući da je reč o standardnom video formatu, slike se mogu snimati i direktno na video rikorder.

Sa stanovišta fotografa elektronske kamere pružaju neke veoma zanimljive prednosti u odnosu na film. Osetljivost najboljih među ovim kamerama dostiže vrednost od ISO 3200 (36 DIN), odnosno pet otvora blende više u odnosu na klasične filmove koje većina ljudi obično koristi i čija osetljivost se kreće oko ISO 100 (21 DIN), čime se postiže visoka efikasnost u slabim svetlosnim uslovima. Osim toga, budući da je po prirodi „elektronska", osetljivost takve kamere se može sasvim slobodno menjati po celoj skali osetljivosti, što fotografu daje potpunu slobodu da prema aktuelnim svetlosnim uslovima bira odgovarajuću osetljivost. To znači da praktično ne mora da spušta kameru sa očiju da bi promenio film, niti da brine o otvoru blende i brzini ekspozicije, čija kombinacija daje često dragocene efekte. Temperatura boje se može menjati po volji – filmovi, naročito dijapozitivi koji koriste profesionalci, se proizvode u varijantama prilagođenim dnevnoj ili veštačkoj svetlosti – čime se eliminiše još jedan razlog za promenu filma ili bar pažljivo i komplikovano filtriranje.

Jedan od razloga za povećanu svetlosnu osetljivost elektronske kamere su dimenzije CCD elementa, čija površina iznosi oko 4.5 cm^2, što je relativno malo u poređenju sa dimenzijama slike od 36×24cm na 35 mm filmu. CCD element pokriva samo jednu petinu slike na 35 mm filmu, tako da se kompletna slika može fokusirati na znatno manji radijus, čime se dobija na intenzitetu svetla. Zahvaljujući ovoj činjenici, profesionalne elektronske kamere imaju još jednu prednost: koriste iste objektive kao i klasične foto kamere (i „Sony" i „Nikon" koriste standardni „Nikon"-ov bajonet-fasung). Pošto objektiv sada fokusira sliku na manju površinu, efektivna žižna daljina raste, tako da osrednji zum-objektiv montiran na CCD kameru bez ikakvih izmena postaje veoma snažan zum. Budući da su standardni snažni zum-objektivi veoma teški, glomazni i skupi, sasvim je očigledno kolika je prednost u pitanju. Međutim, to je koren i jedne mane – širokougaono snimanje iz istih razloga postaje veliki problem.

...UVEK I KUSO?

Najveći problem elektronskih kamera je kvalitet slike – ne postoji jednostavan način uspostavljanja elektronskog standarda za rezoluciju koji bi efikasno i sigurno broj piksela pretvarao u fotografsku rezoluciju (fotografska rezolucija, odnosno *moć razdvajanja*, zavisi od mnogih faktora, a najviše od zrnatosti filma, i izražava se u linijama po milimetru). Rezolucija standardnog 35-mm filma iznosi oko 60 linija/mm (60 parova crnih i belih linija) – ako se uzmu po dva piksela za svaku liniju, odnosno par crnih i belih linija, ovakvoj rezoluciji bi odgovaralo 3000 x 4300, odnosno 13 miliona piksela. Ovo je veoma blisko rezoluciji od 4 000 linija koja se koristi u većini kompjuterskih uređaja za prenošenja slika na slajdove, ali i drastično manje od 150 000 elemenata koliko sadrži video slika. Pored toga, veliki uticaj na kvalitet ima i proces razlaganja na nijanse sivog, što se najočiglednije može videti upoređivanjem slika dobijenih ručnim skenerima sa 256 nijansi sivog sa istim slikama iste rezolucije izražene u tačkama po inču.

Profesionalni fotografi vrlo retko koriste sliku onako kako je „uhvate" – najčešće se prave srednji ili veliki izrezi (uzimanje dela slika), koji se zatim uvećavaju, tako da rezolucija mora biti dovoljno velika da ne utiče previše na kvalitet uvećanog dela slike. Profesionalna „Sony"-jeve i „Nikon"-ove kamere postižu rezoluciju od 380 000 piksela, što ih čini idealnim za primenu u elektronskim medijima, mada ne i dovoljno dobrim za štampu. Zbog toga ovakve sisteme koristi uglavnom samo jedan broj TV fotoreportera.



HDTV UTICAJI

Po svemu sudeći, video će opet odigrati ključnu ulogu – tehnologija HDTV (*High Definition TV*) video kamera će pre ili kasnije biti prihvaćene i primenjena u elektronskim foto kamerama. Jedan od prvih sistema koji će se približiti kvalitetu približnom onom kod filmova će, paradoksalno ali i logično, doći iz „Kodak"-a. „Kodak" veoma pažljivo prati šta se događa u oblasti elektronskih slika, budući da će potrošačka elektronika po svemu sudeći potisnuti film, kao što su kamkorderi na kraju potisnuli klasične 8 mm amaterske filmske kamere. „Kodak"-ov *Digital Camera System* (DCS), za sada namenjen profesionalnim foto-reporterima, sagrađen je oko „Nikon"-ovog F3 tela sa CCD jedinicom i baferom koji može da prihvati nekoliko kadrova. Kao masovna memorija se koristi *Digital Storage Unit* sa hard diskom, koji može da prihvati 150 nekomprimovanih ili oko 600 komprimovanih slika, mada ova kompresija neće biti potpuno transparentna.

CCD elemente sa rezolucijom od 13 mikrona piksela, kao zasebne uređaje za crno-belo ili kolor snimanje, razvio je sam „Kodak". To je veliki korak napred kojim se nova tehnologija sasvim približila kvalitetu potrebnom za novine, mada je još daleko od kvaliteta potrebnog za ilustrovane časopise. Ipak, nisu svi korisnici zainteresovani za tako visok krajnji kvalitet – na primer, korisnicima PC računara je sasvim dovoljno odstizanje kvaliteta njihovih monitora i/ili štampača. Na vrlo sličan način su i ručni skeneri, mada daleko slabijeg kvaliteta slike od stonih A4 modela, našli put od kupaca koji su imali potrebu za unosom ilustracija u dokumente, ali nisu želeli ili mogli da plate cenu velikih uređaja.

Elektronska fotografija predstavlja veoma važan deo novog tržišta, budući da omogućava gotovo trenutno snimanje i prikazivanje slike. Upotrebom elektronskih kamera umesto skenera, slike mogu da se integrišu u novine ili bilo koji dokument bukvalno u trenutku.

KA ZAJEDNIČKOM STANDARDU?

Slike za dokumenta, baze podataka i prezentacije mogu da prođu sa rezolucijom od 100-200000 piksela. Vrlo često su crno-bele slike sasvim dovoljne, budući da su kolor štampači, naročito oni kvalitetniji, još uvek prilično retki. To znači da je postojeća tehnologija već vrlo blizu zahtevima tog segmenta tržišta. Elektronske kamere za kompjutersku fotografiju se već mogu nabaviti po ceni od ispod 1 000 GBP. Prva je bila „Canon"-ova kamera nazvana *Ion*, a sada i „Logitech" pušta u prodaju svoju digitalnu kameru nazvanu *Fotoman*. Ova dva modela se dijametralno razlikuju i po pristupu i po korišćenoj tehnologiji. *Ion* je u suštini analogna video kamera sa ugrađenim diskom za smeštanje slika. CCD panel i procesor proizvode Hiband (profesionalni standard) PAL video sliku u odnosu koje odgovara SVGA standardu.

Slika dobijena elektronskom foto-kamerom ima manje-više isti format kao i slika dobijena visokokvalitetnom video kamerom uperenom u nepokretni objekat. Za unos u računarsku aplikaciju, slika treba da se konvertuje u digitalni oblik pomoću posebne kartice u jednom od PC slotova. U principu, tehnologija nije revolucionarna, ali ono što ovo dostignuće čini izuzetnim je sabijanje optike, opto-elektronike i diska u kompaktnu kameru koja se može lako držati jednom rukom. „Logitech"-ov *Fotoman* proizvodi digitalnu sliku unutar same kamere pomoću A/D konvertera koji je smešten iza CCD elementa. *Fotoman* zapravo proizvodi dve slike: jednu pune veličine i pune rezolucije i drugu manju, veličine poštanske marke, koja služi za brz pregled i identifikaciju slika koje su već smeštene u memoriju kamere. Slika se komprimuje u odnosu 1:9 upotrebom verzije JPEG standarda, tako da u 1MB interne memorije kamere mogu da stanu 32 slike. Cene i mogućnosti nabavke memorije su bili glavni faktori koji su usporili razvoj potpuno digitalnih kamera i ograničili postojeće modele na crno-bele slike.

Slično „Logitech"-ovom *Scanman*-u, i *Fotoman* raspolaže sa 256 nijansi sivog, a skaliranje se dinamički menja za svaku novu sliku. Pre snimanja, kamera uzorkuje svaki četvrti piksel na svakoj četvrtoj liniji CCD elementa, uzima najvišu vrednost i proglašava je za 255: na taj način kamera samu sebe podešava na najbolje moguće iskorišćenje svih 256 nijansi u zavisnosti od toga da li je slika svetla ili tamna. Pošto su slike već digitalizovane, može im se pristupiti jednostavnim priključivanjem kamere na slobodni RS 232 port na računaru. Digitalna slika se zatim obrađuje u posebnom programu koji radi pod *Windows*-ima.

Problemi prouzrokovani velikim brojem različitih formata slika – počev od PCX pa nadalje – bi mogli da se izbegnu još na nivou digitalnih uređaja za smeštanje slike. Preko 100 kompanija radi na utvrđivanju zajedničkog standarda koji će biti primenljiv u svim uređajima, programima za obradu slika i platformama. Prvobitno akronimovan kao CLASP, a zatim preimenovan u TWAIN, novi standard je podržan od firmi kao što su „Aldus", „Caere", „Kodak", HP i „Logitech" (*Fotoman* već generiše slike u skladu sa novim standardom). Ako se novi standard konačno usvoji, većina uređaja kao što su skeneri, kamere ili tzv. *Frame Grabber*-i će moći da se koriste sa bilo kojim odgovarajućim softverom na bilo kojoj mašini bez potrebe za dodatnim hardverom ili posebnim drajverima.

Prema: *Personal Computer World*
Priredio: *Marko Kirić*

ŠTA IMA NOVO

Windows

WINDOWS 3.1 U PRODAJI

Početak prodaje „Microsoftovog" paketa *Windows 3.1* bio je najavljen za 6. april i u „Microsoftu" su odlučili da obezbede maksimalni promet – samo u SAD „Microsoft" je obučio i obezbedio dovoljnom količinom informacija više od 8.000 prodavaca i službenika koji rade na korisničkoj podršci. Pored toga, aktivirana je i prodajna mreža širom sveta, tako da se pojava novog paketa na istočnoevropskom tržištu (u koje spadamo i mi) očekuje najdalje krajem aprila (obično je potrebno da prođe najmanje šest nedelja od izlaska na američko tržište da bi se neki proizvod pojavio u Istočnoj Evropi). Da bi se ovako ambiciozan cilj ostvario, „Microsoft" je obezbedio najveću logističku operaciju koju je do sada preduzeo. Devet proizvodnih tačaka širom sveta će za ovu priliku raditi svih 24 sata dnevno, a tom prilikom će se upotrebiti osam miliona disketa. Procenjuje se da će količina gotovih paketa samo za američko tržište biti dovoljna da popuni 107 ogromnih kamiona transportera, a u pripreme su uključeni i takvi detalji kao što je prilagođavanje parkinga uz proizvodna mesta 24-časovnom radnom vremenu.

Windows 3.1 sadrži više od 1 000 poboljšanja u odnosu na verziju *3.0* koje ga čine bržim, lakšim za upotrebu i čvršćim. Do kraja prošle godine je prodato više od devet miliona kopija paketa *Windows 3.0*, i cilj ovakve operacije je da obezbedi kontinuitet prodaje novog paketa. Tokom rada na novoj verziji sondirano je tržište kroz anketiranje nekoliko hiljada tipičnih korisnika, čiji su odgovori uzeti u obzir prilikom daljeg rada, a na kraju je angažovano 15 000 beta testera. Novi paket će se istovremeno pojaviti na šest jezika: engleskom, francuskom, nemačkom, španskom, italijanskom, holandskom i švedskom.

Testovi

BENCHMARK SARADNJA

"Microsoft" i „Ziff-Davis Publishing Company" su najavili planove za kreiranje *Winstone Benchmark* softverskih testova dizajniranih specijalno za merenje performansi računara koji rade pod *Windows* operativnim sistemom. Testovi će emulirati tipične korisničke aplikacione scenarije, a razvijaće ih „Microsoft" i „PC Magazine" (koji izdaje pomenuta izdavačka kuća), koristeći sugestije iz industrijskih izvora. Testovi će biti široko distribuirani proizvođačima personalnih računara i ostalima koji provode nezavisne testove.

Winstone Benchmark testovi će testirati zbirne performanse PC-ja koji rade pod *Windows 3.X* okruženjem i odgovarajućim aplikacijama. Testovi će biti bazirani na određenom broju unakrsnih tabela, tekst-procesora i ostalih komercijalnih programa koji rade sa fajlovima različitih veličina i tipova, čime bi trebalo da se simulira mešavina zahteva tipičnog PC korisnika. *Winstone Benchmark* testovi će predstavljati komplement ostalim sličnim testovima kao što su *PC Magazine Labs* testovi i kolekcija testova Kompanije za performanse poslovnih aplikacija (*Business Application Performance Company* – BAPCo), čiji su aktivni članovi i „Intel", „Microsoft" i „Ziff-Davis". Pošto će testovi biti sadržajno intenzivni, biće distribuirani na CD-ROM diskovima.

OKLOPNJAČA PANASONIK

Prošla su vremena kada su 24-iglični štampači predstavljali luksuz – poplava jeftinih laserskih štampača ih je načinila standardom i, na radost kupaca, dovela do oštre konkurencije. Sam kvalitet otiska više nije bio dovoljan, pa su u igru ušli fontovi, hardverska oprema, brzina, formati i, naravno, cena, tako da su za 9-iglične modele dani po svemu sudeći izbrojani.

Marko Kirić

Kad se pomene firma „Panasonic" prva asocijacija je najčešće video rikorder ili televizor. Međutim, u računarskom svetu se ova firma poslednjih godina pročula po ambicioznom nastupu na tržištu matričnih štampača. Modeli ove firme su vrlo brzo izašli na glas kao pouzdani, kvalitetni i prilično brzi. „Panasonic"-ove štampače kod nas prodaje beogradska firma „Ibis Sys" (nekadašnji „Ekonomski biro") koja nam je za ovu priliku ustupila novi 24-iglični model formata A3 sa oznakom KX-P1540.

PRVI UTISAK

Prvi kontakt je bio istovremeno i impresivan i zamoran – nikada nisam imao previše poverenja u lagani hardver, a ovaj štampač je sa svojih 15,8 kg sve pre samo ne lagan. Šlednja nikad nije bila odlika ozbiljnih firmi, a „Panasonic" se potrudio da u svakom pogledu ostavi utisak ozbiljnosti, uključujući i pakovanje dostojno oklopnjače. Za raspakivanje je bilo potrebno mnogo više vremena nego za priključivanje – štampač je prvo trebalo izvaditi iz kutije i stiroporne obloge, ali pravi posao je tek tada nastupio. Skidanje raznih zaštitnih omotača, raspakivanje vođica za pojedinačni papir i (metalnog!) separatora za perforirani papir, oslobađanje glave i skidanje raznih osigurača svih mogućih pokretnih delova podosta je potrajalo.

Kada je konačno sve bilo na svom mestu, usledilo je detaljno razgledanje i zavirivanje „pod haubu". Kućište je relativno klasično, ravnih i čistih ivica, bez nepotrebnih egzibicija, i veoma masivnog izgleda koji uliva pouzdanje. Plastika je vrlo čvrsta, bez traga ugibanja tako karakterističnog za većinu jeftinih japanskih modela, svetlo-krem nijansa sa tamnosivim dnom i providnim zatamnjenim poklopcem. Vođica za pojedinačni papir je neobično čvrsta, sa mehaničkim graničnikom koji je obezbeđen izvanrednim klik mehanizmom i, što je posebno zanimljivo i retko viđeno, ugraviranim lenjirima za sve veličine slova (10, 12, 15 i 17 cpi). Jedina zamerka koja bi se mogla staviti ovom inače sjajno napravljenom delu je relativno mala visina – papir se lako previja preko ivice.

Ispod poklopca (koji je, uzgred budi rečeno, snabdeven mikro-tasterom čija je uloga sprečavanje štampanja dok se poklopac potpuno ne zatvori) ukazala se zaista impresivna glava za štampanje veličine pesnice smeštena na nešto manje ubedljiv metalni nosač koji klizi po solidnoj metalnoj poluzi debljine palca. Uopšte, u celom štampaču ima puno metala, uključujući osovinu masivnog gumenog valjka, separatora za pojedinačni papir, vodilica za ugrađeni potisni traktor, pa čak i poluge za određivanje debljine papira. Šasija je posebna priča – napravljena je od milimetarskog lima i celom štampaču daje izgled uređaja koji se kupuje jednom ili dva puta u životu. Utisak je donekle pokvarila minijaturna kaseta sa tankom trakom koja putuje zajedno sa glavom za štampanje. Jeste da je ovakav tip kasete sve češći, ali mi to ipak ni najmanje ne uliva poverenje. Zaista je prava šteta što ovakva glava za štampanje nije snabdevena i odgovarajućom trakom, na ravno onakvom kakvu koristi i „Epson" za sve izuzev kolor modela. Zamerku (opet sasvim uslovno) zaslužuje i zupčasti kaiš za pogon glave koji, mada ništa manji nego i u većini ostalih štampača ove klase, u ovakvom tenkovskom okruženju deluje nekako slabašno, kao da je tu samo privremeno.



Snimio: Nenad Marjanović

Štampač iz gornje srednje klase: Panasonic KX-P1540

Lična karta

Dimenzije (ŠxVxD) 610 x 155 x 390 mm
Težina 15,8 kg
Broj iglica 24
Radni vek glave 200 miliona udaraca po iglici
Bafer 18 KB, proširivo do 32 KB
Papir
- Pojedinačno od 102 x 127 do 419 x 363 mm, 53-90 g/qm
- Traktor do A3 (maksimalna širina do 394 mm), 61-75 g/qm

Veznici Centronics, RS 232
Emulacije
- Epson LQ 1500
- IBM Proprinter
- Diablo 630

Tekstualni modovi
- Draft (10, 12, 15, 15, 20/17 cpi)
- Tekst (10, 12, 15, 15, 20/17 cpi)
- LQ (Courier) (10, 12, 15, 15, 17, 20 cpi)

Korisnički definisani fontovi
- Draft (do 256 karaktera)
- LQ (do 161 karaktera)
- Kombinacija do veličine bafera

Slovne matrice
- 24 x 9 tačaka (draft)
- 18 x 18 tačaka (tekst)
- 24 x 30 tačaka (LQ)

Grafika
- 8-iglična, maksimalno do 240 x 180 dpi
- 24-iglična, maksimalno do 360 x 180 dpi

Brzina
(Deklarisana/izmerena, pri 10 cpi)
- Draft 200/118 cps
- Tekst 115/90 cps
- LQ 67/47 cps

KX-P1540 je standardn
opremljen i paralelni
(CENTRONICS) i serijs
232 interfejsom, tako
dovoljno samo priklju
odgovarajući kabl.

KX-P1540 je standardn
opremljen i paralelni
(CENTRONICS) i serijs
232 interfejsom, tako
dovoljno samo priključ
odgovarajući kabl.

KX-P1540 je standardn
opremljen i paralelni
(CENTRONICS) i serijs
232 interfejsom, tako
dovoljno samo priključ
odgovarajući kabl.

Odličan u tekstu: Tri kvaliteta ispisa

U sličnom maniru je napravljeno i uputstvo za upotrebu koje jeste da nema sjaj i šminku kakvi su sada u modi (štampano je u čistoj crno-beloj tehnici, na običnom papiru većeg formata), niti je napisano na četiri, šest ili desel jezika, već samo na jednom (u ovom slučaju nemačkom), ali zato sadrži sve, ali zaista sve potrebne informacije, uključujući i detaljne tabele karaktera, uputstva za kreiranje korisnički definisanih karaktera i opise svih *Escape* sekvenci. Zaista je osvežanje ponovo videti ovako detaljno uputstvo – nije sasvim kao „Epsonovo", ali nije ni mnogo daleko; u svakom slučaju jedno od najfunkcionalnijih koje sam imao prilike da vidim tokom poslednjih godinu-dve dana.

PODEŠAVANJE I UPRAVLJANJE

KX-P1540 je standardno opremljen i paralelnim (CENTRONICS) i serijskim RS 232 interfejsom, tako da je dovoljno samo priključiti odgovarajući kabl. Podešavanje se obavlja pomoću tri bogata seta DIP prekidača smeštenih na zadnju ploču: prvim setom se postavljaju standardne opcije, kao što su tip emulacije, preskakanje perforacije, način pomeranja papira i nacionalni setovi znakova. Na raspolaganju su tri tipa emulacije: Epson LQ 1500, IBM Proprinter ili Diablo 630 (!), dva seta grafičkih karaktera i 11 nacionalnih setova znakova, među kojima naravno nema našeg. Ovaj nedostatak je ispravljen tako što zastupnik ugrađuje u ROM štampača YU-set, a na raspolaganju je i kompletan ćirilični ROM, što je detalj za svaku pohvalu.

Drugim setom DIP-prekidača se podešavaju parametri vezani za rad sa papirom i baferom – širina (A4 ili A3), način uvlačenja pojedinačnih listova papira (manuelno ili automatski), uloga bafera (*download* ili samo bafer za štampanje) itd., a treći set služi isključivo za podešavanje parametara serijskog interfejsa.

Na prednjoj i gornjoj ploči štampača se nalaze poluge za preklapanje između pojedinačnih listova i beskonačnog papira i uvlačenje pojedinačnih listova, kao i kontrolni panel. Kontrolni panel je priča za sebe – jednostavno ne znam šta da mislim o njemu. Smešten je, naravno, na prednju ploču, malo nakosen, kako to već nalažu zakoni ergonomije, ali su „tasteri" membranski, sa štampanom folijom u boji nalepljenom odozgo i sa gomilom raznobojnih LE dioda. Svi tasteri imaju dvostruke funkcije, a svaka funkcija je propraćena odgovarajućom kombinacijom jedne ili dve LE diode (ukupno ih ima devet) koje ne samo što svetle u svim mogućim kombinacijama, već u određenim slučajevima i žmirkaju. Sve u svemu, prosečan korisnik treba dobro da namuči i sebe i knjigu da bi pohvatao sve konce.

Da ne bi bilo nesporazuma, kontrolni panel nije loš, daleko od toga, čak je i bogatiji od mnogih koje sam imao prilike da vidim, ali jednostavno nije na nivou mehanike i kvaliteta štampača u celini. Ono što posebno otežava ocenu kontrolnog panela je rotacioni preklopnik kojim se reguliše dužina papira i kojim se mogu podesiti dužine od 4, 5, 5.5, 6, 7, 8, 8.5, 11, 12 i 14 inča – izvanredna ideja koju bi mogli mirne duše da iskoriste i ostali proizvođači, uključujući i „Epsone". Naime, perforirani beskonačni papir je najčešće dužine 12", a pojedinačni listovi su najčešće dužine 11", međe nije retkost sresti i drugačije vrednosti. Stvar se posebno komplikuje na A3 štampaču kod koga je moguće ubacivati A4 listove i poprečno, a DIP-prekidači na poleđini većina štampača obično pružaju samo izbor između 11 i 12" i to uz mnogo neprijatne gimnastike za one koji često menjaju papir. Pomalo iznenađuje što projektanti nisu isti princip upotrebili i za izbor fonta i vrste karaktera.

RAD SA PAPIROM

Sasvim u skladu sa onim što sam od štampača ovog formata i kvaliteta očekivao, KX-P1450 „guta" skoro sve što se može zamisliti, kako pojedinačne listove formata od 102 x 127 mm do 419 x 363 mm, tako i perforirani papir širine od 102 do 394 mm. Proizvođač preporučuje papire od 61 do 75 gr kao optimalne, ali se štampač nije zbunio ni sa papirom od 100 gr, kao ni sa višestrukim kopijama. Korišćenje višestrukih kopija je i predviđeno, pa je štampač opremljen regulatorom sa šest položaja, što je veliki plus za sve korisnike koji često rade sa višestrukim kopijama.

Za rad sa pojedinačnim listovima je zadužen masivni gumeni valjak i poluga za pritiskanje papira koja u kombinaciji sa ručicom na gornjoj ploči služi istovremeno i kao komanda za uvlačenje. Papir se jednostavno namesti na vodicu, poluga za pritiskanje odvoji i ručica povuče za još jedan korak, čime se aktivira mehanizam za uvlačenje. Uvlačenje je prilično precizno, ali se stiče utisak da bi preciznost bile još veće da je inače izvanredno dizajnirana vodica malo duža. Ovakvom mehanizmu se ne može ništa zameriti sa stanovišta funkcionalnosti, izuzev činjenice da je papir nemoguće namestiti na sam vrh – uvek nedostaje taj jedan inč na početku koji mora da legne ispod poluge za pritiskanje, a ista primedba važi i u slučaju korišćenja perforiranog papira. Sam red se odvija potpuno neprimetno, što znači i



Sjajan u grafici: Savršeni krugovi

bezbolno za korisnika – za sve vreme testa nije bilo ni jednog slučaja pogrešnog uvlačenja ili gužvanja papira, kako pojedinačnog, tako i beskonačnog. Traktor je bio onoliko pouzdan koliko se i može očekivati od potisnog tipa – u toku testa ni jednom nije zakazao i svoj posao je obavio stoički i bez zamerke. Pri radu sa beskonačnim papirom je veoma bitno da se efikasno razdvoje ulazna i izlazna traka, a za to su konstruktori našli prilično originalno i elegantno rešenje. Umesto jednostavnog spuštanja vodilice za pojedinačne listove, ovde se koristi poseban metalni (žičani) separator u vidu posebnog dodatka.

DOBAR TEKST...

Što se kvaliteta otiska tiče, tu ovaj štampač dolazi do punog izražaja – na raspolaganju nisu dva stepena kvaliteta otiska, kao što je to uobičajeno, nego tri. Tu su grubi (i brzi) draft sa slovnom matricom od 24 x 9 tačaka, zatim mod nazvan *Text*, koji raspolaže matricom veličine 18 x 18 tačaka i nalazi se negde između klasičnog draft i LQ otiska na većini ostalih 24-pinaca, i konačno LQ mod sa matricom od 24 x 30 tačaka. Kombinacija jeste neobična, ali i vrlo delotvorna – koliko puta ste samo pri štampanju nekog listinga ili read.me fajla uz neki program poželeli da štampanje bude malo brže, makar i slabijeg kvaliteta ili vam je pak LQ način bio nedovoljno kvalitetan bez obzira na brzinu? Na raspolaganju je samo jedan ali zato vredan font u veličinama od 10 do 17 cpi i u proporcionalnoj verziji: slova su čvrstog i ubedljivog oblika, lako čitljiva i sa zaista izvanredno oblikovanim YU-karakterima. Izmerena brzina pokazuje sasvim uobičajena odstupanja od deklarisane, što ne znači da je štampač spor, već da jednostavno svi proizvođači navode veoma optimistička cifre, pri čemu naravno ne uračunavaju vreme potrebno za pomeranje papira. Upravo podatak o vremenu pomeranja papira predstavlja podatak po kome mogu da se raspoznaju ozbiljni proizvođači – retko ko ih navodi, izuzev kad želi da se time pohvali. U vrlo iscrpnom spisku tehničkih podataka nalazi se i ovaj o prosečnih 100 ms uz pretpostavljeni prored od 4.2 mm.

Bafer od 18 KB (koji se može proširiti do 32 KB) omogućava učitavanje do 256 draft ili 181 LQ korisnički definisanih znakova, a moguće je i kombinacija od recimo 128 + 128. U principu, moguće je učitati bilo koju kombinaciju draft i LQ korisnički definisanih znakova koji

Nastavlja se na strani 53

SSSSSS OVE STRANE DUGEEEEEE!

U pretprošlom broju „Računara" smo prvi put predstavili jedan kolor skener – Hewlet Packard ScanJet IIc. Pošto smo izazov koji je nudio ovaj uređaj shvatili veoma ozbiljno, preduzeli smo niz opsežnih (i skupih!) testova da bismo utvrdili do koje mere ovakav skener može konkurisati profesionalnoj opremi.

Zoran Životić

U međuvremenu je u redakciju na test stigao sličan uređaj, ali nešto drugačijeg kalibra – SCANMASTER 3+ kolor skener je po nekim svojim karakteristikama uveliko opredeljen za profesionalnu DTP primenu. Pored prikaza ovog uređaja dajemo i neke od rezultata testova sa komentarima koje smo izvršili sa HP skenerom.

SCANMASTER 3+

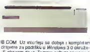
Pakovanje SCANMASTER 3+ kolor skenera firme Howtek je onakvo kakvo bi i očekivali za uređaj koji skenira površine do formata A3 – sa svojim dimenzijama 74 x 51 cm skener deluje kao kompaktnija foto-kopir mašina, posebno zbog izuzetne visine od 24 cm. Sa ovakvim dimenzijama ni težina nije iznenađenje – čitavih 30 kg. Uređaj ne samo po spoljnim dimenzijama već i na osnovu pogleda u unutrašnji mehanizam, po svemu odaje utisak robusne sprave spremne da izdrži mnogo časova rada.

SCANMASTER 3 je klasičan takozvani „flat-bed" skener kod koga se uzorak postavlja na gornju ravnu staklenu površinu i pritiska plastičnim poklopcem. Omogućava skeniranje u opsegu rezolucija od 30 do 400 tačaka po inču. Preciznost A/D konvertera je 8 bita za svaku od osnovnih boja, što znači da je skener u stanju da formira 24-bitnu kolor digitalizovanu sliku, prepoznajući tako preko 16 miliona boja. Interfejs prema računaru je poznati GPIB (General Purpose Interface Bus) i to u širini od 8 bita – o protoku podataka se brine posebna kartica za ovaj interfejs koju smo dobili uz skener.

Softversku podršku skenera nudi firma Howtek svojim proizvodima *Scan II* i *MacScan II*, ali smo u paketu dobili softver nezavisne firme *Mathematica Tempra Pro* – program namenjen grafičkom dizajnu koji uključuje direktnu podršku više različitih skenera, među kojima je i Howtekov model. Uz skener dobijate skromno uputstvo (A4 format u plastičnom registratoru) koje pokriva samo instalaciju i održavanje mašine, kao i kratko uputstvo za instalaciju dodatne opcije za skeniranje slajdova koju smo takođe dobili na test. Uz program *Tempra* ide klasično štampano uputstvo od nekih dvestotinjak strana.

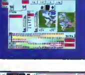
...I ŠTA SVE MOŽE

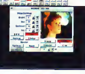
Prvi korak pri instaliranju skenera je ubacivanje GPIB kartice u računar. Ova operacija na dobro „napunjenom" PC računaru nikada neće proći bez problema, pa smo i ovaj put propisno odradili analiziranje posebne dokumentacije uz GPIB kartu (20 strana), probe sa raznim adresama, usaglašavanje podešenog hardvera sa softverom itd. Ipak, uz pomoć priloženog programa za dijagnostiku pronalaženje prave kombinacije je znatno olakšano – kada je program jednom potvrdio ispravnost instalacije, skeniranje iz programa *Tempra Pro* je proradilo „iz prve". Instalacije softverske podrške GPIB interfejsu za DOS okruženje se svodi na instaliranje rezidentnog drajvera GPIB.COM. Uz interfejs se dobija i kompletan set drajvera za podršku u *Windows 3.0* okruženju. S obzirom da je *Tempra* softver namenjen čistom DOS-u, nismo imali prilike da probamo *Windows* instalaciju – uputstvo nas uverava da je o njoj vođeno dovoljno računa, jer se spominje i *Windows 386 enhanced* mod, mogući problemi s njim i način na koji se mogu rešiti.



Snimio Nenad Marijanović

Skeneru je nakon uključivanja potrebno dvadesetak sekundi da se inicijalizuje i bude spreman za rad. Režim u kome se nalazi se može pratiti preko četiri indikatora na prednjem panelu koji ukazuju na stanja „spreman", „zauzet" i „greška". Ovo su ujedno i jedini indikatori – skener nema nikakvih drugih kontrola na panelu, jer se sve obavlja programom za skeniranje.





Poslednji Mohikanac: *Temora Pro* radi pod DOS-om i, ignorišući *Windows* standard, dolazi sa sopstvenim grafičkim interfejsom

Način skeniranja je određen samim programom *Tempra* i izveden je na manje-više klasičan način. Program nudi podešavanje osnovnih parametara (rezolucije, oštrine, kolor balansa itd.), a zatim i opciju probnog skeniranja, izbora površine i finalnog skeniranja. Brzina skeniranja se kod ovog uređaja mora prikazati preko dva parametra – vremena potrebnog za pripremu i samog skeniranja. Prvo vreme je, bez obzira na zadate parametre, uvek približno isto – potrebno je oko 30-40 sekundi da skener uopšte počne sa radom. U zatvorenom paketu skener+softver kao što je ovaj teško je detektovati šta je uzrok ovako dugačkom vremenu pripreme. Način na koji se šalju komande skeneru, potreban „overhead" itd su nepoznati, ali ipak imamo utisak da se krivica ravnomerno može rasporediti i na softver i na hardver.

Samo skeniranje će za uzorak formata 9 x 12 u punoj 24-bitnoj kolor rezoluciji potrajati oko 30-tak sekundi za rezoluciju 100 x 100, s tim što sa povećanjem finoće skaniranja ono drastično raste. Za vreme skeniranja uređaj prilično čeka i glava se kreće u sitnim koracima. S obzirom da je propusna moć GPIB interfejsa, bar prema deklaraciji, oko 200 K/s, svu krivicu za ovakav rezultat snosi u manjoj meri softver, i u mnogo većoj meri računar. Iako je za vreme testa korišćen dobro konfigurisan 386 računar, očigledno je da PC klasa još nije spremna da korisnicima pruži brzu manipulaciju slikama pri ovakvim rezolucijama i sa ovom količinom kolor informacija. S obzirom da smo isti utisak imali i kod testiranja HP skenera, ostaje nam jedino da zaključimo da je mogućnost skenera očigledno ispred mogućnosti računara, te da sa ovom opremom nismo u stanju da proverimo stvarnu maksimalnu brzinu.

Svakako da posebnu vrednost ovom skeneru daje mogućnost skaniranja slajdova. Da bi to bilo moguće, potrebno je nabaviti dodatnu opremu koje se sastoji iz posebnog panela za osvetljavanje sa gornje strane. Uklanjanjem plastičnog poklopca skenera (jednostavno, uz pomoć dva vijka sa nazubljenim glavama koji se mogu odvrnuti rukom bez alata) sa strane se pojavljuje široki prorez po kome se kreće nosač ove konzole vezan za interni pokretni mehanizam u samom skeneru. Na taj način slajd biva prosvetljen sa gornje strane. Uz ovaj uređaj dobija se i nekoliko foto-kartona sa prorezima u standardnim dimenzijama slajdova (od formata 6 x 6 na više), kojima treba pokriti celu gornju površinu ostavljajuć da samo slajd bude izložen svetlu. Postavljanje slajda na ovaj način, vodući pri tome računa da bude postavljen tako da linije budu horizontalne, zahteva malo veštine i prakse. Dodatno je obezbeđena i difuzor ploča koja omogućava ravnomernija osvetljavanje sa gornje strane.

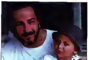
TEMPRA PRO

Program *Tempra Pro* kojim se obavlja skeniranje nije namenski program već opšti alat za razvoj profesionalnih publikacija i prezenta





Original, onako kako ga vidi BIGZ-ov skener u rezoluciji 53 linije po centimetru (132 tačke po inču), sa kompletnom pripremom filmova na profesionalnim grafičkim uređajima (na slici autor teksta Zoran Živolić sa svojim sinom)





Reprodukcija, onako kako je vidi HP ScanJet IIc u rezoluciji 40 linija po centimetru (100 tačaka po inču), sa pripremom filmova na uređaju Linotronic 330 (Studio B & Z)

cionog materijala. Njegov prikaz bi zahtevao tekst za sebe, pa ćemo ovde zadržati samo na nekim osnovnim karakteristikama.

Ono što će vas svakako iznenaditi je da se radi o programu koji radi isključivo pod „čistim" DOS okruženjem i koji sve svoje zadatke u grafičkom prikazu obavlja sam. Ovakva koncepcija je danas, kada je *Windows* okruženje već standard za grafičke aplikacije, zaista retkost i izaziva određene nedoumice kod korisnika o potrebnom nivou prilagođavanja korisničkom interfejsu. *Tempra Pro* zaista ima svoj interfejs, ali se on, sem u dizajnu i to na svoju štetu, u velikoj meri poklapa sa onim na šta samo navikli kod *Windows*-a. Tu su i padajući meniji i komandna dugmad i *scroll bar* ovi i sve ona kontrola koje inače poznajemo. I dizajn ima nekoliko istih „linesa", što se pre svega odnosi na senke koje odaju utisak treće dimenzije, ali sve zajedno deluje nekako grublje. Logika rada sa mišem (potreban spoljni drajver) i način aktiviranja opcija je, u osnovi sličan *Windows*-u pa ćete ipak osnovne operacije moći da obavljate vrlo brzo. Ipak, bez pomoći uputstva nećete moći da se snađete, jer na mnogim mestima zalaskom u pojedine opcije prestaje mogućnost intuitivnog upravljanja programom na osnovu Windows iskustva.

Sve mogućnosti programa možemo sažeti u jednu reč – kompletno. *Tempra Pro* poseduje sve opcije koje se danas smatraju standardnim kod grafičkih paketa gornje klase – od osnovih operacija sa linijama i površinama, preko manipulacija tekstom u grafičkom obliku, do finih filtriranja slike i raznih efekata. Program podržava rad sa slikama u različitim formatima – tu su TIF, TGA, GIF, PCX i *Windows* format. U stanju je da radi sa različitim količina kolor informacije – do 1 bita (crno-belo) pa sve do 32 bita, što je već retkost kod ovakvih programa.

Tempra Pro odaje utisak izuzetno koncepcijski doslednog i razrađenog programa. Brzina programa je sigurno veća nego da je realizovan pod *Windows*-ima i, prema našoj proceni, dovoljna je da ne unosi nepotrebne prekide u rad. Programom smo radili na 386 konfiguraciji sa izuzetnim video sistemom (Turbo Mega EVA sa rezolucijom 1024×768 i 16000 boja na 20-inčnom monitoru – test oba uređaja ćete pronaći u ovom broju) na koma je pokazao izuzetne rezultate u vernosti prikaza kolor fotografije. Ipak, nadamo se da firma Mathematica planira i *Windows* verziju ovog programa, jer nam se čini da sem upotrebe uz skener sa kojim čini dobru celinu, sam program neće steći mnogo pristalica ukoliko i dalje bude bežao od standarda – ma koliko nam se ovaj standard sviđao ili ne.

NA TESTU KVALITETA

Fotografije koje su pred vama su u potpunosti pripremljene za štampu primenom personalnih računara. Svi uzorci su korektno urađene kolor fotografije formata 9 x 12 cm, koje su skanirane HP ScanJet IIc skenerom. Skeniranje je vršeno bez ikakvih korekcija boja ili osvetljenja u rezoluciji 100×100 tačaka po inču (kvalitetna kolor štampa ide do 60 linija po centimetru, odnosno 150 tačaka po inču), prihvatanjem inicijalnih parametara koje nudi HP program. Binarne datoteke su u TIF formatu prenete na mekintoš računar na kome je, uz pomoć programa *Adobe PhotoShop* izvršeno razdvajanje boja, a zatim uz pomoć programa *Quark Express* izvučeni filmovi na Linotronic 330 osvetljavajućoj jedinicom (ide do 2540, a korišćeno 120 tačaka po inču). Postupak je obavljen u firmi Studio B & Z. Filmovi su direktno predati štampariji BIGZ-a i rezultat celog postupka je pred vama. Iako Linotronic 330 sa rezolucijom do 2540 tačaka po inču iscrtava tačku sa izvanrednom preciznošću, stručnjaci u BIGZ-u su primetili da su ivice ovih tačaka previše oštre (profesionalni skener crta tačke sa mekšim ivicama), što daje tvrđu reprodukciju boje. Linotronic, naravno, i nije namenjen za kolor pripremu, već za osvetljavanje teksta, a tu se neophodne upravo tačke oštrih ivica.

Rezultati koje vidite su nekoliko koraka udaljeni od maksimuma koji je moguće dobiti ovakvim postupkom. Ne znamo doduše kako će ispasti sama štampa, ali je na osnovu probnog otiska bilo jasno da je fotografijama ipak neophodna korekcija osvetljaja za oko 15-20%. Balans boja je potpuno zadovoljavajući iako nismo vršili kalibraciju skenera. Blago tačkasti efekat koji se primećuje je posledica ipak nisko zadrane rezolucije – povećanjem na 160 tačaka po inču efekat je potpuno nestao.

Profesionalci koji su analizirali filmove su potvrdili da se je ceo postupak izveden maksimalno korektno, sem pomenutog osvetljaja. Separacija boja koju je obavio Adobe program i način na koji su formirane tačke na filmu (kod kolor sepracije koristi se lezni pomak boja – svaka boja koristi tačke orijentisane pod određenim uglom kako bi nakon nanošenja boje u štampi svaka boja zauzela jedan segment kruga koji se tretira kao tačka) su urađene potpuno u skladu sa profesionalnim zahtevima.

Ohrabreni ovako dobrim rezultatima, pokušali smo da ceo proces izvedemo i sa daleko jeftinijom tehnologijom – filmove i Linotronic 300 smo zamenili laserskim štampačem i pausom. Jasno je bo početka bilo da dobijeni rezultati neće biti u domenu profesionalnih, ali nas je pre svega interesovalo da li je ovakva kombinacija uopšte moguća. Boje su razdvojene *CorelDraw*! programom koji ima ovakvu opciju i potom 4 „filma" odštampana kao PostScript datoteke. Otisak je izvučen na pausu uz pomoć HP LaserJet II sa ugrađenom kartom za povećanja rezolucije na 600 x 600 tačaka po inču (DP-Tek LaserPort). Rezultat je u najmanju ruku interesantan. Kao što smo i očekivali rezultat nije upotrebljiv (na taj način su pripremljena slika ekrana uz prikaz HP skenera), ali razlog nije ni u rezoluciji ni u korišćenom softveru za razdvajanje boja, već prosto u tehnologiji otiska. Laserska tačka na pausu je ipak suviše neprecizna da omogući pravilnu orijentaciju i oštrinu tačke. Iako da se umesto popunjavanja kruga dobijaju nepravilni preklopi koji uništavaju kvalitetnu kolor prezentaciju. Naravno, problem je pojačan deformacijama koje paus doživljava pri grejanju u laserskom štampaču. Tako da čak i lista sa bo-ovojenim bojama pokazuju dodatnu neusklađenost.

Na kraju ostaje nam da nakon svega zaključimo kako dva kolor skenera koje smo testirali omogućavaju kolor pripremu (HP foto grafija, a Howtek i fotografija i slajdova). Izrada filmova za kolor pripremu i dalje ostaje u domenu strogo profesionalne opreme, ali se jedna takva usluga srećom može iznajmiti i kod nas. Naravno, kolor priprema nije jedina, pa možda ni prevashodna namena kolor skenera ove klase. Svoju osnovnu primenu oni će verovatno naći u dokumentarističkim poslovima – unos raznih dokumenata u računar – a tu je Howtek sa A3 formatom i mogućnošću skeniranja materijala na transparentnim podlogama u izvesnoj prednosti. Arhivarski poslovi su međutim sasvim druga priča.

Korisne adrese
Osa (Howtek) Mice Stojkovića 8, 11130 Beograd Tel. 011/557-592, Fax 011/559-845
Penhard (Hewlett Packard) Ivana Milutinovića 24, 11000 Beograd Tel. (011) 436-019, Fax (011) 435-513

PLOTER NEŽNOG LIKA

Ploteri velikih formata spadaju u najkompleksnije i najskuplje delove hardvera CAD sistema. Razlog njihovoj ceni, pa time i nedostupnosti širem krugu korisnika je veoma skupa mehanika na kojoj se oni zasnivaju i čija cena praktično ne pada već godinama. U ovom tekstu prikazaćemo vam jedan od vrhunskih A0 pen plotera – 1835 CR firme „OCÉ Graphics".

Nebojša Novaković

U ovom trenutku tržište plotera velikog formata (A1, A0 i veći) podeljeno je u tri klase: pen, ink jet termalni i elektrostatički. Pen ploteri, odnosno crtači sa perima, predstavljaju klasiku u ovoj oblasti. Oni sadrže karusele u koje možete postaviti određeni broj različitih pera, flomastera, specijalizovanih hemijskih olovaka ili čak grafitne mina – sve to sa različitim bojama i debljinama. Uglavnom šta su da su glava plotera ih prihvata iz karusela i vuče ih po medijumu bilo u samo jednoj dimenziji (ako ploter pomera medijum po drugoj – roll bed ploteri) ili u obe dimenzije (ako je papir statičan – flat bed ploteri). Najbitnija odlika pen plotera je to što je njihov način crtanja vektorske prirode – jednako kao i kod CAD paketa. Ako je u crtežu, na primer, kosa duž pod nekim uglom zadata sa svojom početnom i krajnjom tačkom, pen ploter će jednostavno povući pero (i papir) između te dve tačke i dobićemo pravu liniju jednaku onoj koju bi dobar tehnički crtač iscrtao.

Kod druge dve vrste plotera – ink jet i elektrostatičkih (i najnovijih laserskih) situacija je međutim sasvim drugačija. Naime, to su rasterski ploteri koji, kao i ink jet ili laserski štampači, iscrtavaju sliku kao matricu tačaka, tako da kod kosih duži i krivina nazubljenja postoje. Takođe, vektorski crtež se mora najpre rasterizovati da bi ga ovakav ploter iscrtao, s tim što taj posao sada pretežno automatski obavljaju brzi RISC procesori u samim ploterima.

S druge strane, ovi ploteri su kod svih složenijih crteža znatno brži od pen plotera, jer je mehanička brzina štampanja (pošto je u principu jednak proces kao i kod štampača) manje-više konstantna od 3 minuta po A0 stranici (kod thermal ink-jet) sve do 10 A0 stranica u minuti (kod laserskih). Prednost u brzini dolazi do izražaja naročito kod više kopija istog crteža. Kod pen plotera brzina iscrtavanja zavisi od kompleksnosti crteža, najviše od broja i dužine vektora. Medijum kod današnjih plotera inače može biti hartija u više tipova i debljina, paus i folija.

Najpoznatiji proizvođači plotera velikih formata su američki „Calcomp", „Hewlett-Packard" i „Houston Instruments", japanski „Roland", „Mutoh" i „Graphtec" i nekoliko evropskih proizvođača, među kojima je i „Oce Graphics", čiji ploter prikazujemo u ovom tekstu.

OCÉ G1835-CR

Francuska firma „Oce Graphics" je deo velike firme „Oce" – poznatog proizvođača kopir aparata i drugih kancelarijskih mašina. „Oce Graphics" nastao spajanjem „Oceovog" tima za računarske periferije i poznatog „Bensona" svojim asortimanom proizvoda pokriva oblasti plotera, digitalizatora i kolor termalnih štampača. Serija 1800 sadrži pen plotere A1 i A0 formata. Ploter G1835 CR, koji smo testirali, pripada srednjoj klasi „Oce Graphics" A0 plotera.

G1835 CR pripada klasi „roll bed" pen plotera, tako da bez obzira na veliki format papira zauzima relativno malu površinu. U gornjem delu su nosač papira, glava za pero i karusel kao i sva elektronika i interfejsi. Sve to nosi pijedestal visine reda xx cm. Ukupne dimenzije plotera su X x Y x Z cm.



Ovaj ploter, kao i većina drugih u ovoj klasi, ime svoj 16-bitni mikroprocesor koji nadzire njegov rad, ali posebnost G1835 su njegova puna četiri megabajta RAM namenjena kao bafer za smeštanje plot fajlova. Zahvaljujući ovako velikoj lokalnoj memoriji, računar može poslati ne jedan, nego nekoliko crteža u ploter i nastaviti s radom dok ih ploter redom crta.

Proizvođač ističe još nekoliko prednosti ovog plotera, kao što su:

- Pen Manager, koji proverava i zamenjuje pera pre nego što im se tuš istroši
- Area Manager, koji optimizuje putanju i zamenu pera sortiranjem vektora, ubrzavajući tako crtanje
- Curve Manager za umekšavanje krivih
- Soft Landing mehanizam za maksimalan kvalitet sa tehničkim perima
- inteligentni kontrolni panel sa menijima na više jezika (ne i našim, na žalost) i 2 x 20 LCD ekranom za izbor naredbi i prikaz poruka
- Setup Help funkcija, koja omogućava prepoznavanje i snimanje parametara crteža

G1835 CR je dosta dobro opremljen po pitanju veze sa računarima. Tu su dva RS 232 C serijska porta i jedan opcioni Local Talk port.

KONTROLNI PANEL

G1835 CR spada u grupu „ljubaznijih" plotera, mada je sada nivo mogućnosti upravljanja manje-više jednak kod svih plotera visoke klase. Na gornjem delu ploter je grupa tastera sa malim LCD ekranom. Ovaj LCD, sa 2 reda po 20 znakova, u gornjem delu pokazuje poruku o trenutnom stanju plotera, a u donjem poziciju dugmeta kojim se startuju meniji za rad sa ploterom. Naredbe iz ovih menija omogućuju da se najveći deo operacija koje su bile vezane za softver, kao obrtanje, skaliranje, pomeranja crteža, crtanje „u ogledalu" ili crtanje više kopija, obavljaju neposredno na ploteru. Veoma lepa mogućnost je korisničko definisanje usmerenosti X i Y osa preko 2 ili 3 tačke, tako da po želji možemo menjati ugao referentnih koordinatnih osa i na taj način deformisati crtež. Druge zanimljive mogućnosti su automatsko odsecanje delova crteža koji su van papira i automatsko prepoznavanje vrste i debljine pera (ili drugog medijuma, pod uslovom da je Oce).

Na kontrolnom panelu je i poseban taster na kome je simpatično iscrtan zec u trku. To je TURBO taster s kojim se ploter prebacuje u najbrži način rada. Brzine su tada do 100 cm/s (65 cm/s normalno) za crtanja sa hemijskim olovkama ili 30 cm/s za rapidografe srednje debljine (0,25 – 0,5 mm). Kod crtanja tankim perom 0,18 mm nema turbo načina, a brzina tada je svega 10 cm/s, jer sa većom takvo pero ne bi opstalo više od nekoliko sekundi.

UTISCI U RADU

Bez obzira na svega nekoliko sati koliko smo imali vremena da praktično isprobamo ploter, rezultati su bili dosta dobri. Rad sa jednom rolnom papira A0 širine i dužine i do 50 metara, veoma velika, reklo bi se čak nervozna brzina crtanja u turbo modu i izuzetna inteligencija ostavile su izvanredan utisak. Interesantno je pogledati kolika je bila praktična razlika u brzini crtanja istog crteža perom 0,35 i „hemijskom olovkom". Ono što pero iscrta za 8 minuta i 40 sekundi hemijska olovka crtala je za nešto preko 4 minuta i 50 sekundi – skoro dvostruko ubrzanje. Naravno, treba uzeti u obzir i određeni gubitak kvaliteta u bržem slučaju.

Bez obzira na relativno nežan izgled, barem u poređenju sa recimo HP Draftmaster II, G1835 prilikom plotovanja proizvodi veoma male vibracije, izuzev na 100 cm/s u turbo modu. Tada ovaj ploter, koji je inače vanserijski tih, postaje vanserijski bučan, što jeste određeni minus. S druge strane, nijedan veliki ploter na brzinama iznad 80 cm/s nije tih.

Sve u svemu, „Ocaphics" 1835 CR spada u grupu plotera koji će svojom brzinom i kvalitetom zadovoljiti svaku inženjersku firmu. Toj grupi korisnika ovakav izlazni uređaj ne samo što se veoma brzo isplati, nego je i obaveza u određenom smislu. Naravno, za slučajeve ekstremno velikih količina crteža, tu su uvek termalni, elektrostatički ili laserski ploteri, ali je tada i cena „odgovarajuća".

Korisna adresa

Osa
Mice Stojkovića 8, 11130 Beograd
Tel. 011/557 592, fax. 011/559-845

MANJI OD KNJIGE

Evo jednog zaista zanimljivog proizvoda u klasi sa kojom se do sada nismo bavili – tzv. „booksize" (veličine knjige) računara. Mala kutija, pristojna cena, zgodan, sladak – ali, može li da izdrži pritisak konkurencije?

Dejan V. Veselinović

Nema ni jednog suštinskog razloga zbog koga bi računar morao biti veliki, sem ostavljanja na volju vlasniku da ga po želji gledi i dograđuje. Već na današnjem nivou minijaturizacije sasvim je moguće napraviti veoma malo sistemsko kućište, a da se pri tom ne žrtvuje nijedna bitnija radna osobina mašine.

Ovakav stav je danas ostvaren u nekoliko mašina koje se prodaju u inostranstvu, ali na i kod nas. Bilo je nekoliko pokušaja da se ovakvi računari ponude i kod nas (recimo „Peripheral") ali izgleda da tržište nije bilo spremno za ovakve mašine.

Pa ipak, evo jednog novog pokušaja. Pred nama se našao jedan od dve osnovne mašine ovog tipa, iz koje je izveden niz modela za manje ili više posebne namene.

KAKO IZGLEDA I...

Izgleda stvarno kao knjiga osrednje veličine 285×220×50 mm (ŠxDxV). Uz njega dobijate i smanjenu tastaturu slične veličine i smanjeni crno-beli monitor sa dijagonalom ekrana od 9 inča (22 cm). Na prvi pogled, delovi neodoljivo podsećaju na dečju igračku, na neki računarčić za klince i klinceze. Međutim, kada ga pažljivije pogledate, shvatite da on nije ni najmanje naivan. Naprotiv!

Prvo tehnologije platforme koristi procesor 80286 na 16 MHz bez stanja čekanja. Veznu logiku obezbeđuje Headland HT12 VLSI čip, a o video podsistemu se stara jedan „Oak" C67 video čip sa 512 KB video memorije, što je dovoljno za rad u rezoluciji od 1024×768 sa 16 boja ili 800×600 sa 256 boja. Na matičnoj ploči se nalaze integrisani jedan AT Bus adapter, kontroler flopi diskova i cela I/O sekcija. Ukratko, sve što biste i očekivali od jedne AT mašine, pa još malo više, jer Super VGA još uvek nije standard.

A ni veznika ne nedostaje. Na poleđini čak i ovako malog kućišta naći ćete veznike za VGA monitor, tastaturu, paralelnu i serijsku vezu, čak i liniju za igre (game port). Opet, kao i pravi veliki računar.

Usled osetno smanjenog unutrašnjeg prostora arhitektura mašina više liči na „Compaq" proizvode nego na uobičajenu tajvansku robu. Izvor napajanja se proteže od prednje do zadnje strane kućišta, veoma je uzan i ide od vrha do dna, a zbog malih dimenzija ventilator za izduvavanje toplog vazduha je mali isto kao i na prenosivim mašinama. Usled malih dimenzija kapacitet izvora napajanja je jedva 60 VA, mada je to objektivno gledano sasvim dovoljno za ono što je u računaru i što se može dodati. Lepa osobina napajanja je što se samo prilagođava naponima u mreži od 90 do 260 V.

Tastature veoma podseća na prenosnike – smanjena je, pa je moralo doći do preklapanja funkcija na tasterima. Iako je očigledno da je ova tastatura napravljena kao prilikac računaru, i uprkos činjenici da objektivno nije loša, ne bismo vam je preporučili – zašto biste se mučili sa raznim kompromisima ako ne morate? A ne morate, možete upotrebiti sasvim normalne AT tastature. Ova je smanjena samo zato da bi se mogla spakovati u najlonsku torbu zajedno sa računarom, čime se dobija lako prevoziva mašina, ako vam to nije bitno, kupite odmah neku tastaturu standardne veličine.

Što se mehaničke robusnosti samog kućišta tiče, dobija čistu desetku – mali je, ali je žilav, da ne kažemo pegav.

...I KAKO RADI

Shodno protokolu, proterali smo ovog švrću kroz celokupan niz mašinskih i programskih testova. Dobijeni rezultati sasvim potvrđuju naša očekivanja, a ponegde ih i prebacuju.

U testovima procesora OTC se ponašao sasvim nalik referenci, što je i bilo očekivano, jer koristi potpuno istu tehnologiju kao i naša referenca. Međutim, u testovima tvrdog diska i video podsistema veoma osetno je prebacio referencu. Disk „Seagate" ST3144 je dao indeks od 1.21, a poredio se se nimalo naivnim „Quantum" modelom LPS52A, ovo se mora smatrati čak odličnim rezultatom. Ako je on odličan, rezultat na video podsistemu, sa indeksom od 1.72, se mora smatrati izuzetno dobrim. Rezultat je inače dobrog „Oak Technologies" OTI067C video čipa, koji je samo potvrdio razloge zbog koga ga cenimo (jeftin a dobar).

Ukupni prosečni indeks na mašinskim testovima je odličnih 1.23, što nije ni malo čest slučaj.

Ovaj niz uspeha se nastavio i na programskim testovima, mada na nešto nižem nivou. Od pet testova, OTC je nadmašio referencu u četiri, a u petoj je zaostao jedino zato što referenca koristi numerički koprocesor (IIT 80C287), koga kod modela OTC nije bilo (iako, naravno, postoji podnožje za njega). Čak i sa tim zaostajanjem (indeks 0.8), prosečni indeks brzine je dobrih 1.05, a ako izuzmemo test sa koprocesorom, indeks bi iznosio 1.11. Možda 11% brže nije neka novost, al' za iste pare, pa još u manjoj kutiji, pa šta ćete više.



Za specijalne namene: OTC booksize

Snimio: Nenad Marijanović

ČEMU SLUŽI...

Ovakva koncepcija sama o sebi otvara pitanje kome i za šta može zatrebati ovakva mašina. Nije reč o tome da je mnogo lep – mada je naša bolja polovina kada ga je ugledala uzviknula „Jao, što je sladak!"

Računar je mali i izuzetno pogodnog oblika da ga negde gurnete i mahene zaboravite, sem kada vam zatreba disketni pogon. Uprkos maloj dimenziji, sposoban je da primi flopi disk jedinicu, tvrdi disk koji ako ima format od 2.5 inča (recimo, modeli firme „Conner" sa kapacitetom od 40/60/80 MB), može da se postavi ispod flopi disk jedinice, pa to omogućava i dalje raspolaganje sa jednom utičnicom za proširenje. Nju opet možete upotrebiti za faks/modem karticu ili mrežnu karticu (u kom slučaju bi makar faks mogao da bude povezan na mrežu).

Dakle, ako ste korisnik pojedinac kome treba čista funkcija i koji nema namere da bog zna kako širi računar, ovo je mališan idealna mašina. Sa malim postoljem postavite ga uspravno pored monitora i uživajte.

S druge strane, ako vam treba mrežni terminal, ovaj mališa vam može lepo odraditi taj posao, opet zato što je mali pa ne zauzima prostor na kancelarijskom stolu, na kome ionako nikada nema dovoljno mesta. Pored toga, mehanički je natprosečno otporan, pa ga možete i klepiti bez ikakvih posledica.

U vezi terminala isto važi i za njegovu tastaturu i monitor, koji su takođe smanjeni – opet borba za dragoceni kancelarijski prostor.

I naravno, tu su i deca. Ako planirate nekom detetu da kupite računar, ovo je veoma ozbiljan kandidat po praktično svemu, a pre svega opet mehaničkoj robusnosti.

...I DA LI VREDI

I tako dolazimo do pitanja cene. U OTC-u su nam rekli da će ova mašina sa pomenutim delovima koštati nešto malo više od slične stone mašine. Recimo, monitor treba da košta koliko i svaki drugi monohromatski VGA monitor – taj deo računice nam nije sasvim jasan, po nama bi on trebalo da košta makar nešto malo manje. Isto važi i za tastaturu – valjda zbog relativno malih serija i ona će koštati nešto više od standardnih stonih (jeftinijih) modela, a jedina prednost koju realno nudi je manja veličina.

S druge strane, ništa vas ne obavezuje da kupite baš te delove celine, pa se razlika u ceni može delimično smanjiti.

Poslednji aspekt o kome treba razmisliti je primena ovog modela kao nekog poluprenosnog računara. Naime, računar i tastatura se mogu spakovati u priloženu torbicu koja ih čini lako prenosivim, ali neupotrebljivim ako na drugoj strani puta nemate monitor (odnosno ako ne ponesete onaj mali). Baš te mogućnosti čini ovaj model veoma interesantnim za pisce koji polovinu vremena provode na jednom, a polovinu vremena na drugom mestu, pa im treba lako prenosiva mašina. Za takve korisnike, ovo je bez ikakve sumnje jeftinije rešenje od pravog prenosivog modela, koji bi sigurno koštao daleko više.

Za ostalo i dalje stoji činjenica da se očekuje da će ova mašina koštati nekoliko procenata više od drugih sličnih računara na koje smo navikli. Da li se to isplati ili ne, teško je reći, naš stav je da se isplati pod uslovom da nemate nameru da bog zna kako širite ovaj računar. Jedino, pazite da pri kupovini tvrdog diska odmah nabavite mali disk koji vam neće uzeti prostor za proširenja, sem te opaske, ako niste skloni brzim promenama, savetovali bismo vam da obavezno pogledate ovaj model.

Slika 1



Mašinski testovi

Korisna adresa

OTC
Sava Centar, Milentija Popovića 9
11070 Novi Beograd
Tel. (011) 2222-355, 2224 322/323 i 684
Fax (011) 2222-355

VIDEO KARTICE TURBO MEGA EVA/1023, ATI VGA WONDER, DIAMOND SPEEDSTAR SUPERVGA

KRALJICE LEPOTE

Do sada smo imali posla uglavnom sa „normalnim" VGA video kartama. Vreme je da malo pažnje posvetimo i modelima koji su za sada još uvek retkost, ali će u najskorije vreme postati sasvim standardna stvar. To su VGA karte sa 33.000 pa i više boja, na kojima je sve učinjeno da prikaz slika bude što brži a njen izgled što verniji.

Dejan V. Veselinović

Odavno se zna da je glavni problem na VGA karticama njihov RAMDAC – odnosno digitalno-analogni konvertor. To je čip čija je dužnost da digitalni signal pretvori u analogni i kao takvog ga pošalje monitoru. Uprkos činjenici da ove čipove pravi čitav niz firmi po celom svetu, od kojih su najpoznatije i najcenjenije „Inmos" (Velika Britanija, vlasništvo italijansko-francuskog koncerna „SGS-Thompson") i „Brooktree" („BI" SAD), oni još uvek nisu dovedeni u fazu „dovoljno jeftino – dovoljno brzo". Naime, DAC je glavna „kočnica", odnosno često najsporiji deo lanca obrade video signala, daleko sporiji od samog video čipa.

ŠESNAESTOBITNA KONVERZIJA

Analogni monitori ne boluju od boljke zvane „ograničenost boja" od koje su patili digitalni monitori – mogu doslovno da prikažu neograničen broj boja, koliko god im se signalom uputi. S druge strane, video čipovi današnjice, pogotovo svi savremeniji modeli, takođe nemaju bitnijih ograničenja u smislu broja boja, sem čiste količine memorije i manjih problema koji se mogu rešiti izmenama u BIOS-u. Prema tome, jedini razlog zbog koga smo predugo gledali 16 boja iz palete od 64 (VGA standard) ili u najboljem slučaju 256 boja iz palete od 16,7 miliona jeste taj nezgodni DAC čip.

Drugi problem koji je delimično vezan za DAC je „reckavost" slova na ekranu. Naime, u veoma velikom broju slučajeva stanje između dva susedna piksela je moglo da bude samo „da" (osvetljen) ili „ne" (zatamnjen), bez ikakvih međuvarijanti. Jedan način da se sa ovim problemom izađe na kraj jeste da se podigne opšti nivo rezolucije, čime se problem veštački prekrive usitnjavanjem tačaka i povećanjem njihovog broja. Što vizuelno dobrim delom rešava problem, ali vlasnika baca u trošak za skuplji monitor. Opet, dok se stvar rešava rezolucijom, sve je to ipak samo polurešenje. Nama treba ono što se zove „anti-aliasing", odnosno uvođenje polustanja, čime se sa nepromenjenom rezolucijom dobija mnogo ispravniji oblik slova.

Odnedavno, nakon dugog vremena i nesumnjivo mnogo rada, na tržištu su se pojavili DAC čipovi koji umesto dosadašnjih u najboljem slučaju 8-bitnih režima (tj. 256 boja) mogu da rade u 15-bitnom režimu, a to znači istovremeno sa 32.768 iz bar isto toliko velike palete. Jedan od tih novih modela ima i pomenuti „anti-aliasing" efekat ugrađen u hardver, pa se faktički istovremeno rešavaju dva značajna problema (Edsun CEG čip).

O tom drugom DAC čipu ćemo reći više kada nam dođe do ruku, za sada, pozabavili bismo se onim koji nam jeste u rukama, a to je čip sa oznakom SC 11486CN američke firme „Sierra". Probali smo tri kartice, jednu dobro poznatu i dve koje su, koliko znamo, novitet na našem tržištu. Zanimalo nas je da ih uporedimo sa „standardnim" proizvodima u uobičaje-



Snimio: Nenad Marjanović

Turbo Mega Eva/1023

NOVE TEHNIKE

U TEORIJI I PRAKSI

Postoje u osnovi dva načina da se ubrza prikaz slike na ekranu. Prvi je da se upotrebi metod poznat kao „frame buffering", ili, u slobodnom prevodu, međumemorija za slike, a drugi, naravno, kartica sa sopstvenom memorijom i video procesorom kao što je TI 340XX serija; ovo drugo je već pravi pravcati računar na kartici, što jasno pokazuju i da velikom cenom.

Kartica koju smo dobili na probu, koja koristi nama do sada potpuno nepoznat NCR video čip sa oznakom 77C22, koristi prvi metod. Pored standardne video memorije od 1 MB na kartici se nalazi i dopunski megabajt memorije koja služi za čuvanje niza podataka o tome šta tačno treba uraditi. Proizvođač ovu memoriju naziva „extended" (produženom) memorijom, nalik svemu iznad 1 MB na AT mašinama.

Po teoriji, kada se od video karte očekuje da izvrši čitav niz radnji ili da ponovi neku radnju niz puta, svi podaci sa tim u vezi bi trebalo da se brzo prebace iz sistemske u ovu posebnu memoriju na samoj video kartici, a rezultat bi trebalo da bude veća brzina rada, odnosno izvršavanja. Međutim, prečesto se dešava da je teorija jedno, a praksa nešto sasvim drugo. Najčešće veće performanse vuku i veću cenu, pa je zato veoma oportuno uporediti dobiti sa razlikom u ceni.

Ova kartica bi trebalo da košta oko DEM 300 u Minhenu, odnosno oko 30% više od, recimo, „Tseng" ET4000 kartice. Još jedan elemenat koji treba imati u vidu jeste da sva rešenja oko upotrebe ove dopunske memorije moraju biti data od strane samog proizvođača, bilo u hardveru, bilo u softveru, jer se radi o nestandardnom režimu rada. Drugim rečima, ova kartica se mora probati sa posebnim veznicima koji se uz nju dobijaju.

U najboljoj tradiciji američkih proizvoda sve što uz karticu dobijate je zaista prvoklasno. Ovde se posebno misli na vezne programe, koji su potpuno automatizovani u smislu instalacije (kao i kod WD 90C30), ali i na razne korisničke programe, kojih ima od sta i koji su veoma dobro urađeni i odabrani. Odmah vidite da imate posla sa izuzetnim proizvodom.

S druge strane, na kartici koja inače ima izvanredno kvalitetnu štampanu pločicu, utisak kvare tri otpornika koji su naknadno zalemljeni; ovo je pouzdan znak da se radi o prvoj generaciji proizvoda, a da će ovakve mane biti otklonjene u drugoj iteraciji. Druga stvar koja pomalo kvari utisak je zabuna oko položaja četiri prekidača: u priručniku piše jedno, a u praksi je upravo obrnuto. Oni kažu nadole, a u praksi je nagore. Još jedna dečija bolest.

Iako smo mi dobili model sa standardnim RAMDAC čipom, sposobnim da prikaže samo 256 boja, postoji i verzija sa „Sierra" čipom i 33.000 boja.

U tabeli 1 ćete lako primetiti da je ova kartica pokazala osrednje rezultate u tekstualnom režimu rada, ali da je zato osvetlela obraz na testovima pod *Windows*-om. Ovo ju

Tabela 1

		ATI	Tseng	Diamnond	C&T	NCR
Tekst režim						
• BIOS poziv bez skrolovanja		37 44	24 38	447 16	40 12	23 11
• BIOS poziv sa skrolovanjem		15 77	13 68	30 52	8 97	14 01
• Neposredno na ekran 8-bit		797 40	812 60	1005 00	298 00	797 00
• Neposredno na ekran 16-bit		1570 40	1591 20	1972 00	594 00	1563 00
• Neposredno na ekran, 32-bit		1728 20	1758 80	1757 80	570 80	1720 00
Grafički režim						
• Ispunjavanje mode 0 8-bit		1626 24	1646 40	1646 60	533 12	1642 45
• Ispunjavanje mode 0 16-bit		2168 32	3277 12	3270 40	577 91	3254 20
• Ispunjavanje mode 0 32-bit		2170 58	3281 60	3277 12	577 92	3260.34
• Memo - Ekran BitBlitter 16-bit		4678 80	6518 40	8087 64	1446 72	6566 40
• Memo - Ekran BitBlitter 32-bit		4986 00	7218 72	7257 80	1430 68	7242 72
• Ekran– Memo BitBlitter 16-bit		9945 60	4545 12	5347 20	1438 80	4651 20
• Ekran– Memo BitBlitter 32-bit		9951 20	4905 60	5013 00	1422 96	4983 00
• Ekran– Ekran BitBlitter 16-bit		10268 18	5000 80	5006 40	2878 40	12910 08
• Ekran- Ekran BitBlitter 32-bit		10272.21	5000 80	5000 80	2872 80	12971.52
Windows testovi (16 boja)						
• BitBlt poravnavanje (128×128 b) prozora		404621	408346	414539	277352	451972
• BitBlt Ekran– Ekran BitBlt prozora		598918	602353	631904	384240	682098
• Pravougaonici (zaobljeni) prozora		1141346	1128278	1140734	975710	1199566
• Fontovi	sivi font znakova	2163	2184	2208	1798	2239
	mešani fontovi znakova	12182	11713	11919	10787	12614
• Opšte	otv /zat. prozora prozora/s	2 58	2.54	2 57	1 93	2 64
	skrol prozora znakova/s	1026	905	924	537	1019
Windows testovi (33 000 boja)						
• BitBlt poravnavanje (128×128 b) prozora		531 138	593982	556766	767204	
• BitBlt Ekran– Ekran BitBlt prozora		658627	740955	694002	1275019	
• Pravougaonici (zaobljeni) prozora		396892	328489	772803	947902	
• Fontovi	sivi font, znakova	836	816	951	872	
	mešani fontovi znakova	5304	5964	6203	7043	
• Opšte	otv /zat prozora prozora/s	1 09	1 42	1 35	2 53	
	skrol prozora, znakova/s	322	354	345	1185	

FRAME BUFFERING

je prirodno učinilo kandidatom za prvo mesto, pa nismo odoleli da je uporedimo sa trenutnim kraljem druma, karticom sa čipom WD 90C30.

Tabela 1 je dobijena u radu na jednom računaru sa procesorom AMD 80386SX na 25 MHz. Međutim, Tabela 2 je dobijena na platformi sa procesorom 80486 na 33 MHz, pri čemu je sabirnica bila navijena ne na 8,3 nego na 13,3 MHz. Ovim smo ujedno ispitali robusnost kartice (radila je bez ikakvih problema) i pokazali kako radi na platformama kojima je suštinski namenjena s obzirom na cenu i namenu (viša klasa korisnika u smislu jačih zahteva).

Rezultati prikazani u Tabeli 2 veoma jasno pokazuju da ova kartica ne nudi bilo kakve prednosti nad WD 90C30 u tri od četiri grupe testova. Pre bi se moglo reći da prilično zaostaje u tekstualnom režimu rada, pa čak i u standardnoj VGA rezoluciji koristeći generičke veznike u radu sa *Windows*-ima.

Pošto ona nije ni namenjena standardnom radu, ponovili smo sve testove sa obe kartice koristeći njihove vezne programe za rad pod *Windows*-om u rezoluciji 800×600 i sa 256 boja (na žalost ne mogu više). Ovaj put dobili smo obrnutu situaciju. NCR je povukao ispred WD-a u šest od sedam testova. U svim slučajevima, radi se o zaista akademskim razlikama od par procenata, sem u dva testa. U prvom je WD povukao za nekih 46% (sivi tekst), a u drugom je NCR povukao za nekih 72% (otvaranje i zatvaranje prozora).

Dakle, stvarne razlike su veoma male, toliko male da bi se mirne duše mogle pripisati kvalitetu veznog programa isto kao i posebnoj memoriji. Da bismo bili zaista sigurni, rezultate smo naknadno proverili i u CAD programu (*DesignCAD 3-D*), u kome smo dobili istovetne rezultate, kao da posebne memorije uopšte ni nema. Čak i pod *Windows*-om u rezoluciji 1024×768 sa 256 boja, odnosi su ostali slični.

Ostaje nam da zaključimo da je u ovom slučaju prednost dopunske memorije prilično mala, mada po svemu sudeći postoji. Relativno je skromna, u proseku oko 10% i to samo na višim rezolucijama, a treba naglasiti i to da ova kartica može da radi do rezolucije od 1280×1024, ali sa preplitanjem. Ima jedna stvar koja se na testovima ne vidi, a to je kontrole slike primera radi gotovo sve kartice koje smo do sada probali su ili ostavljale blago nepodesnu sliku na miru, ili su je još malo krivile. Ova kartica je tu sliku popravila taman toliko da izobličenja postanu manje vidljiva, i to bez upotrebe programskih alatki koje upravo ovome služe a isporučuju se uz karticu.

Sve u svemu, iako nudi samo veoma skromne i po dejstvu veoma ograničene dobiti, pa je žalo ne bismo mogli nazvati korakom napred, ova kartica nam se veoma dopala. Za one koji rade gotovo isključivo u *Windows* okruženju, a posebno sa programima kao što su *Corel Draw*, *Publisher's Paintbrush*, *MS Powerpoint*, pa i *Word for Windows*, razlika u ceni vredi dobiti, ostali neka mirne duše kupe WD karticu, koja ostaje na vrhu u apsolutnom smislu, a obe u verzijama za 33 000 boja.

Uostalom, za nekoliko meseci i neće biti drugog.

nim režimima rada u posebnim režimima rada i, naravno, da pogledamo kolika se cena u performansama plaća za veliki broj boja. Sve kartice podržavaju VESA standard osvežavanja ekrana, sve do 72 Hz.

TURBO MEGA EVA/1023 VGA-SYNC

Gornjem imenu nedostaje samo još atribut poput „Bi turbo GXL Granada Tup-Tup De Luxe" (uzgred, ispod imena zaista još piše i „Multi-color Multi-shade Graphics Adapter" – čudo božje!). Sa takvim imenom može da bude samo sa Tajvana, a kod nas gostuje zahvaljujući novosadskoj firmi „Mikronova".

Tehnički, radi se o generičkom proizvodu firme „Tseng", na kome postoje jasni tragovi prerade za nove standarde. Ovim ne mislimo ništa loše, samo napominjemo da je kartica izvedena iz drugačijeg projekta. Na sreću korisnika, to izvođenje se svodi na izostavljanje niza elemenata koji su vremenom postali suvišni (tu se, pre svega, misli na razne kratkospojnike). Kartica je time postala urednija, mada pomalo nepotrebno velika.

Umesto tri EPROM-a, sada se ceo BIOS nalazi u jednom, pored njega je ROM sa kodom za sinhronizaciju i, najzad, Sierra DAC čip. Memorija se sastoji od osam čipova, nominalne brzine od 70 nS i ukupnog kapaciteta od 1 MB.

Što se prateće opreme tiče, sve je zapakovano u jednu impresivno obojenu kutiju, dokumentacija je dobra, a vezni programi su manje-više isti kao i kod drugih sličnih proizvoda, sem što naravno imaju novije i drugačije veznike. U celini, ovo je i dalje dobar posao, da ne kažemo najbolji na našem tržištu.

ATI VGA WONDER XL

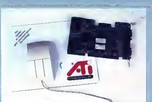
O firmi ATI je svako ko prati zapadnu, a posebno američku, štampu imao prilike da pročita niz hvalospeva. Kod nas, ova firma je poznata kao proizvođač za video podsistem računara „Daewoo", a autor ovih redova poseduje modem ove firme (2400etc) još od 1988. godine, sa kojim je veoma zadovoljan.

Ovaj primerak je vlasništvo firme „Gama Electronics", koja ga je uvezla kao „beta primerak" (odnosno, proba radi), jer planira da počne da nudi baš ovaj model u najskorijoj budućnosti; kada budete držali ovaj broj u rukama, verovatno će već biti u prodaji.

Za razliku od „Tseng" kartice, ATI je ovaj model radio od nule. To se savim lepo i vidi na njoj, pre svega po smanjenim dimenzijama i po malom broju praznih mesta za elemente (često alternativnih oblika). Sve je urađeno po najnovijoj tehnologiji uz punu primenu površinskog lema i za pasivne elemente (otpornike, kondenzatore).

ATI je oduvek voleo da koristi sopstvene video čipove kada god je to moguće, pa je to urađeno i ovaj put. Video čip nosi oznaku „ATI-28800-5", a potpomognut je sa još svega deset čipova, na računajući ROM BIOS i memoriju. I ovde se nalaze svega dva BIOS EPROM-a i, naravno, „Sierra SC11486 DAC". Na samoj pločici su zalemljena dva memorijska čipa za ukupno 256 KB, a ostatak do 1 MB se nalazi u postoljima.

Specifičnost ova karte čini i „bus" veznik za miša koga dobijate uz samu karticu. Miš je, začudo, boljeg kvaliteta od recimo našeg Genix-s GM-6 miša, što iskreno govoreći nismo očekivali, računajući da će nam uvaliti bog zna šta, jer je miš ipak podređen video funkcijama. Kad ono, gle čuda, miš čak vrlo dobar. A ne



ATI VGA Wonder XL



Diamond Speedstar Supervga

traži, ni svoju senjsku liniju niti hleba jede!

Još jedna zaista interesantna specifičnost ove karte je interni sistem nazvan „CODE" (*Color Depth Enhancement* – poboljšanja dubine boje) koji je sastavni deo jednog od veznika za *Windows 3.0*. Elem, taj dodatak omogućava da se slika napravljena u Targa, PCX, TIF i nekim drugim formatima i karticama sa 24-bitnom grafikom (16.7 od 16.7 miliona boja!) pod *Windows*-om gleda sa 262 144 boja, odnosno osam puta više od (novog) standarda ili 1 024 puta više boja od današnjeg standarda.

DIAMOND SPEEDSTAR SUPERVGA

Ovaj proizvod se naveliko reklamira po američkim časopisima kao najbrža VGA karta na svetu, a posebno u radu pod *Windows*-om. Kod nas, ovu karticu prodaje firma „Microair", od koje smo i dobili probni primerak.

U osnovi kartice koristi „Tseng" ET4000 video čip, a sve ostalo je potpisala firma „Speed Star" – i VGA sinhronizacioni modul i BIOS i posebne veznike, ukratko sve sem „Sierra DAC" čipa. Zahvaljujući ovom čipu, kartica je u mogućnosti da upotrebi 32 768 boja. No, iako koristi poznati video čip, izgleda sasvim drugačije od drugih proizvoda na osnovu iste tehnologije, ta činjenica, kao i neka druga rešenja na njoj, nas navode na zaključak da se zaista radi o posebnom proizvodu („Assembled in U.S.A." – Sastavljeno u SAD). Kvalitet rada je dobar, ali ni po čemu izuzetan, pa po opštem utisku zaostaje za ATI karticom.

Za uzvrat, kartica je jednostavna: pored tri EPROM/ROM čipa, video čipa i DAC-a, sadrži svega 7 TTL i 1 PAL čip, kao naravno i osam memorijskih čipova. Polovina njih su zalemljeni za pločicu (512 KB), a druga polovina je u postoljima. Svi su veoma brzi, sa deklaracijom od 70 ns. Na kartici postoje i četiri prekidnika, ali oni oglavnom služe rešavanju problema ukoliko ih bude; mi smo ih proverili, ali nismo imali potrebe išta da menjamo (oni su, inače, dostupni i sa spoljne strana kartice/računara).

Kao ukupni paket-proizvod, „SpeedStar" kartica je u ovoj grupi bez konkurencije, a isto važi i šire. Naime, ima najbolji priručnik na koji smo do sada naišli, kako po organizaciji, tako i po informativnosti. Uz manje-više standardne vezne programe, dobijate i jedan koga do sada nismo vidali, pod nazivom „WINRIX DCS". Autor je poznata kuća „RIX Softworks Inc." koja se bavi grafikom, a koja je inače autor mnogih demo slika koje smo vidali. Ovaj program otvara poseban prozor u *Windows*-ima, u koma se mogu obrađivati slike napravljene u drugim formatima, kao što su Truevision Targa, TIFF, RIX, GIF, BMP i PCX. Pored obrade, ovaj program prebacuje slike iz jednog u drugi format, a ako iz višeg želite u niži, bićete upitani da li i kako želite određivanje boja. Poenta je u tome što ovaj paket obezbeđuje izuzetno celovito rešanje problema velikog broja boja i raznih formata.

NEKI DETALJI

Možda se pitate čemu toliko oduševljenje velikim brojem boja? Vidite, standardna VGA slike u rezoluciji od 640×480 je efektivno bolje rezolucije od TV slike na najboljem prijemniku pod normalnim uslovima, a uprkos tome teško da je stvarna slika bolja. Naprotiv, i golim okom se vidi da je oho-ho lošijeg kvaliteta od TV slike. E, to je isključivo zato što TV radi sa neograničenim brojem boja, a standardna VGA sa svega 16 boja.

Negde oko 4 096 boja se uzima kao neka arbitrazna granica prelaska sa normalne u tzv. duboku boju. 33 000 boja je, prema tome, daleko u domenu duboke boje, a 266 144 je već domen najprofesionalnije tehnike. A sve to kod vaše kuće, na vašem skromnom SVGA monitoru, i to sa rezolucijom koja je nekih 70% bolja od vašeg TV prijemnika. To morate da vidite da biste poverovali, a kako sada stvari stoje, uskoro i hoćete. Za nekoliko meseci to će biti jedine karte na tržištu, jer je već sada razlika u ceni svega DEM 20-30, a brzo je neće uopšte ni biti.

Drugi aspekt koji nas je ovde zanimao, a kojim se do sada nismo posebno bavili, su programski veznici. Sve iznad standardnog VGA režima, odnosno 640×480 sa 16 boja, podra-

Tabela 2

		WDC 90C30	NCR 77C22
Tekst rezim			
• BIOS poziv bez skrolovanja		112.51	27 07
• BIOS poziv sa skrolovanjem		27 84	16 81
• Neposredno na ekran 8-bit		1534 20	1211 80
• Neposredno na ekran 16-bit		2912.80	2318 80
• Neposredno na ekran 32-bit		3739 00	2807 20
Graficki rezim			
• Ispunjavanje mode 0 8-bit		2591 08	2053 83
• Ispunjavanje mode 0 16-bit		4915 07	4092 31
• Ispunjavanje mode 0 32-bit		5176 02	4985 68
• Memo – Ekran BitBlitter 16-bit		11868 64	9718 08
• Memo – Ekran BitBlitter 32-bit		12645 92	11735 04
• Ekran- Memo BitBlitter 16-bit		5959 20	5380 80
• Ekran- Memo BitBlitter 32-bit		7464 96	6543 36
• Ekran- Ekran BitBlitter 16-bit		18063 36	15329 28
• Ekran- Ekran BitBlitter 32-bit		20413 44	18186 24
Windows testovi (16 boja)			
• BitBlt poravnavanje (128×128 b) prozora		719893	683175
• BitBlt Ekran- Ekran BitBlt, prozora		7706873	7702296
• Pravougaonici (zaobljeni) prozora		3464534	3452732
• Fontovi	sivi font znakova	5356	5340
	mešani fontovi znakova	35637	35404
• Opste	otv./zat. prozora prozora/s	6 80	6 33
	skrol prozora znakova/s	1438	1377
Windows testovi (256 boja)			
• BitBlt poravnavanje (128×128 b) prozora		1135716	1285504
• BitBlt Ekran- Ekran BitBlt prozora		1616417	1832170
• Pravougaonici (zaobljeni) prozora		2374620	2745412
• Fontovi	sivi font znakova	3572	2452
	mešani fontovi znakova	12532	15784
• Opšte.	otv./zat. prozora prozora/s	3 70	6.37
	skrol prozora znakova/s	1683	1671

zumeva posebnu programsku pobudu. Program kao program može se napisati dobro i loše, a kako je napisan osetiće se na kvalitetu slike (manje) i brzini rada (više). Drugim rečima, bez obzira na mašinske sposobnosti video karte, veznik može da ih iskoristi ili ne, pa brza karta može raditi veoma sporo, a sporija karta može prestići onu koja je na mašinskom nivou objektivno brža.

I najzad, posebno smo se pozabavili brzinom u radu sa velikim brojem boja i na većoj rezoluciji. Naime, jasno je da ako se od video čipa traži 50% veća rezolucija, pa 128 puta više boja, sve to mora da se odrazi na brzinu. Pitanje je samo koliko, a po nama, odgovor bi mogao da utiče i na izbor platforme.

BRZINSKI TEST

Merenja smo sasvim namerno obavili na jednoj platformi koja koristi AMD 80C386SX procesor na 25 MHz. Ova platforma sada dodaje manje od 10% na cenu ukupne mašine u najmanjoj konfiguraciji i više nema razloga pod kapom nebeskom da iko kupuje AT na 16 MHz. Drugi razlog je bio taj što nas je zanimalo da vidimo koliko će ove video karte subjektivno uticati na brzinu i komfor rada, mada bi po propisima trebalo da obavimo sva merenja na našem poligonu sa procesorom 486, kako bismo u najvećoj mogućoj meri odstranili uticaj mašinske baze.

Pojednostavljenja radi, niz numeričkih podataka smo sabrali u jednu tabelu. „Tekstualni režim" obuhvata klasične testove brzine rada u tom režimu, ali umesto u sekundama izražene u pikselima u sekundi. Naredna kolona „Grafički režim" odnosi se na merenja same kartice na mašinskom nivou kada izvršava grafičke naredbe. Drugim rečima, to je isto kao i grafički režim, ali bez ikakvih veznih programa, a sa ciljem da se ispita mašinska brzina rada u grafičkom režimu. Sve vrednosti su navedene u pikselima u sekundi.

"Windows testovi" su ponovljeni dva puta: jednom u standardnom VGA režimu (640×480, 16 boja) i drugi put u VGA režimu, ali sa svih 33 000 boja. Zato su u drugom bloku testova izostali rezultati za C&T karticu koja, naravno, sve to ne može, a u slučaju rezolucije od 800×600 sa 33 000 boja rezultati su u gotovo istovetnim odnosima, ali niži za 10-12%.



NCR kartica sa baferom za sliku

Lako se može videti da su u tekstualnom režimu „Tseng" i ATI kartica praktično isti, sa tako malim razlikama da nema smisla o njima ni govoriti. Slično važi i za „SpeedStar" karticu, uz nekoliko izuzetaka u oba smera.

U čistom prikazivanju ekrana bez skrolovanja, „SpeedStar" kartica je nadjačala konkurenciju sa faktorom od 12:1 (!). U sličnom testu, ali sa skrolovanjem, razlika u njenu korist se smanjila na gotovo 2:1, da bi na 16-bitnom prislupu pala na 1,23:1 i najzad na 32-bitnim zadacima zaostala za konkurencijom. Sve u svemu, nema diskusije da je u tekstualnom režimu ubedljivo brža od konkurencije.

U grafičkom režimu stvari se već menjaju. U 16- i 32-bitnom ispunjavanju ekrana, „Tseng" preuzima veoma ubedljivo vodstvo nad ATI sa razlikom od oko 50%, a sa nešto manjom razlikom vodi i u testovima prebacivanja slučajno generisanih podataka iz memorije na ekran. U naredne dve grupe testova, prebacivanje podataka sa ekrana u memoriju i sa ekrana na ekran, ATI obrće tablice i vodi se oko 2:1. To znači da ATI kartica daleko efikasnije radi sa memorijom od Tseng-a, u slučaju crtanja celog ekrana ili menjanja velikih delova ekrana, prednost ide onome koji brže radi sa memorijom.

"SpeedStar" se u ovim testovima nije baš proslavila po brzini. Od devet testova, bila je najbrža u svega tri, od čega u dva sa zaista akademskom razlikom.

U standardnom režimu rada pod *Windows 3.0*, od šest testova ATI je poveo u četiri, a zaostao u tri, međutim, radi se o toliko malim razlikama da one zaista spadaju u čisto akademski domen. U praksi, nema šanse da primetite te i takve razlike u brzini. SpeedStar kartica je od sedam testova povela u tri, sa razlikama reda veličine oko 5%, što baš i nije neka zaslepljujuća brzina.

U režimu rada sa velikim brojem boja već dolazimo do veoma subjektivnih stvari, jer se tu u igru mešaju i vezni programi. Od sedam testova ATI je povukao u dva, a zaostao u pet, mada opet sa manje više akademskim razlikama, uprkos većim apsolutnim odnosima, sada već u domenu 10-15%. Jedino je u testu otvaranja i zatvaranja prozora razlika otišla na čak 30% u korist „Tseng" kartice.

Za „SpeedStar" karticu smo odabrali njihov veoma hvaljeni turbo režim rada pod *Windows*-om (mada smo kontrole radi probali i ostale veznike). Oni ga reklamiraju kao „Super fast" (Super brz), a nama ostaje da razmišljamo šta li je toliko super brzo u njemu. Malo smo cinični, jer je istina da je u slučaju testa sa pravougaonicima faktički radila 94% brže od prvog narednog konkurenta. Uopšte, ova kartica je povela u tri od sedam testova, i to simptomatično u testovima sa fontovima. Opet, razlike su na nivou 5%, ali objektivno tu su.

U celini gledano, može se zaključiti da je „Tseng" kartica pokazala manju prednost nad ATI karticom, ali se ta prednost svodi na nivoe koje ćete teško osetiti u praksi kao manju ili veću brzinu same karte. U odnosu na skromnu C&T karticu, razlike postoje, mogu se osetiti i bez štoperice, ali su u većini slučajeva razumno male. Izuzetak je skrolovanje u prozoru, gde je razlika između ove i druge dve kartice gotovo 2:1, a to se lako primeti i golim okom, jer vidite da se kartica nešto muči.

Što se „SpeedStar" kartice tiče, zaista ne vidimo neke velike prednosti koje ona nudi nad konkurencijom, mada ovo kažemo uz jednu

Nastavlja se na strani 23

Tabela 3

Proizvođač Model		ATI VGA	Tseng EVA	Diamond Speedstar VGA	NCR WonderXL 1024
Video čip		ATI 28800	ET4000	ET4000	NCR 77C22
BIOS		ATI/Award	Tseng	Diamond	NCR/Award
Memorija (standard)		512 KB	1 MB	512 KB	1 MB
Memorija (opcija)		512 KB	--	512 KB	1 MB
Standardi	640×480	Da	Da	Da	Da
	800×600	Da	Da	Da	Da
	1024×768	Da	Da	Da	Da
	1280×1024	--	--	--	Da
RAMDAC	16/256 boja	Da	Da	Da	Da
	33 000 boja	Da	Da	Da	--
	66 000 boja	--	--	--	--
	Anti-alias	--	--	--	--
	Posebno (+)	Da		--	--
Posebni režimi					
	a	262K boja	--	--	--
	b	--	--	--	--
VESA standard					
		Da	Da	Da	Da
Veze					
	analogna	Da	Da	Da	Da
	digitalna	Da	--	--	--
	bus (miš)	Da	--	--	--
Veznici					
	Windows 3.0	Da	Da	Da	Da
	OS2 PM	Da	Da	--	Da
	AutoCAD 10	Da	Da	Da	Da
	AutoCAD 11	Da	--	Da	Da
	Ventura	Da	Da	Da	--
	GEM	Da	Da	Da	Da
	WordPerfect	--	Da	Da	--
	Lotus 1-2-3	Da	Da	Da	Da
Ocena		7	8	8	8

(+) Vidi tekst

VIŠE OD STANDARDA

Za novosadsku firmu „Mikronova" znali smo do sada samo iz reklama. Ovo nam ja prvi konkretan dodir sa opremom koju ona prodaje. Ako je suditi po uzorku na testu, neka se ostali prodavci dobro drže.

Dejan V. Veselinović

Radi se o jednom veoma neobičnom sistemu. Prvo video podsistem je van sveta i vremena – 33 000 boja odjednom! I sve to na ekranu sa dijagonalom od 20 inča (51 cm). Kade to vidite, shvatite šta nas čeka sa HDTV (High Definition TV).

VIDEO

Video karta je, na izgled, standardni tajvanski proizvod sa „Tseng ET4000" video čipom. Dve stvari je odvajaju od proseka – prvo, zadržala je prolaznu sabirnicu na vrhu („Pass-through Features Connector") u vidu iglica i drugo, umesto uobičajena tri poveća čipa, dva za BIOS i jedan za RAMDAC, ova karta ih ima četiri! Dva su, naravno, BIOS čipovi, a treći služi kao veza ka četvrtim – „Sierra RAMDAC" (SC 11483 CN), koji omogućava ravno 32 768 boja iz paleta od 16.7 miliona boja odjednom. Dalje, odavno se zna da su RAMDAC čipovi glavna kočnica u brzini rada cele video karte, ali ne i na ovoj kartici. Karta je veoma brza. Naravno, ima 1 MB video memorije.

Što je veoma interesantno, kompatibilna je sa „Targa" formatom, pa su čuvene damo slika bebe i vodopada urađene u ovom formatu. Ovo je veoma važan podatak za sve zainteresovane za animaciju i rad sa velikim spektrom boja (33 000 je već prilično širok spektar) ili sa nedavno prikazanim kolor skenerom HP ScanJetC II koji radi sa svih 16.7 miliona boja. Da ne dužimo priču, ovde radi o 128 puta (!) više boja od današnjih standarda, i to za sasvim malu doplatu u odnosu na standardnu cenu takvih proizvoda (od oko 15-20%).

Monitor je potpuno bezimen – jedina oznaka koju na sebi nosi jeste nalepnica na kojoj piše „Hyperscan". Jeste bombasto, ali je i proizvod bomba. Prvo, radi do rezolucija od 1280×1024 sa i bez preplitanja, a dao bog veliki je, 20 inča po dijagonali, odnosno 51 cm. Pošto su mu sve kontrole sa prednje strane, uvek možete podesiti sliku tako da nema „belina" – crni okvir po ivicama. Slika je stabilna da stabilnija ne može biti, a ne sećamo se da smo do sada videli ovako veliki monitor sa ovako dobrom slikom. „Dobro" u ovom slučaju znači verne boje, stabilna slika, brzi prelazi sa jednog na drugi režim bez problema i mala ugaona izobličenja. Da ne pomislite da je savršen, jer toga nema, kompromis na ceni se vidi u prelasku iz jednog režima u drugi.

Recimo, ako ste u režimu od 600×600 i podesite sliku da zauzima celi ekran, pa onda pređete u bilo koji drugi režim, moraćete ručno da prepodesite sliku, a gotovo je sigurno da će „povući" ovamo ili onamo van vidnog polja. Na skupljim i čuvenijim monitorima, taj posao bi obavio mikroprocesor koji bi zapamtio sve parametre podešavanja za bar 5-6 režima. Naravno, to bi i koštalo više.

DISK

Disk je takođe van standarda. Radi se o proizvodu poznate američke firme „Micropolis", sa oznakom 1664 7. To je ESDI disk kapaciteta od 320 MB, formata 5.25 inča, ali sa pola visine. Doduše, ima solidnu potrošnju od nominalno 20 VA, što je podosta više od recimo 14 VA za „Fujitsu" od 510 MB, ali je još uvek u granicama prihvatljivog. Završna obrada diska zaostaje za sličnim proizvodima iz Japana (poznata priča?), ali ako nema finesu, ima masivnost, pa deluje kao od gromade odvaljen. Disk je dosta tih, ne čuje se mnogo, a ni ne greje se onoliko koliko bi se očekivalo od potrošnje, i to je sasvim u granicama normale.

Povezan je za ESDI kontroler firme „Data Technology Corp" sa oznakom DTC6282-24. Kartica ima ugrađenu keš memoriju i to sa statičkim RAM-om (MOSEL MS62256A 35), a pored zalemljenog čipa postoji i prazno postolje za isti takav čip, koji je očigledno opcioni. Dakle, 32 KB je standard, a može i duplo više. Koliko sve to ima smisla pokazuje jedan podatak – vreme direktnog pristupa, kao i vreme čitanja susednog traga, iznose jedva nešto malo više od 1 ms. Naravno, to je tako u mašinskim testovima, dok u programskim testovima tako mala keš memorija nema baš previše efekta.

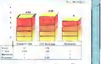
MATIČNA PLOČA

Matična ploča je proizvod poznate i cenjene tajvanske firme „Datatech Enterprises" poznatije pod skraćenicom DTK. Ploča je već zaista tipičan primer proizvodnje ove firme: pune je veličine (to jest, nije „baby" formata), koristi mahom diskretnu tehniku u tehnologiji površinskog lema, sa samo jednim VLSI kolom (VLSI VL82C100) koja kontroliše sabirnicu. Na ploči se nalazi osam postolja za SIM module koji primaju module od 256 KB, 1 MB i 4 MB, što znači da na matičnu ploču možete staviti do 32 MB. Keš memorija može biti 64 KB (kao u našem slučaju) ili 256 KB uz zamenu čipova. Procesor je AMD Am386 DX-40, na radnom



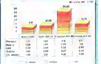
Snimio Nenad Marijanović

Slika 1



Mašinski testovi

Slika 2



Programski testovi

Slika 3



Micropolis 320 MB: Brzine prenosa

taktu od 40 MHz. U našem slučaju, u računaru smo imali montiran i numerički koprocesor Intel 387DX-33. Ovaj spoj nam nije baš jasan: procesor na 40, a koprocesor na 33 MHz, ali da je radilo – jeste. I jedan mali kuriozitet: iako ploča ima DTK BIOS, ovoga puta postoji opcija duplikatne memorije i za sistem i za video, nešto što je dugo bilo izostavljeno.

Poslednja kartica koju treba pomenuti jeste jedna fax/modem kartica, sa oznakom RFAX/modem. Naravno, tajvanske je proizvodnje. Jedino nju nismo testirali u okviru celine, jer ovakve kartice zahtevaju malo dugotrajnije testiranje. Ipak, korektnosti prema prodavcu radi, reći ćemo da na prvi pogled i laganu probu sve lepo radi, a ostaje nam da probamo i druge komunikacione programe.

I SERVER I CAD/CAM

Dakle, imamo računar sa izuzetnim grafičkim mogućnostima, velikim tvrdim diskom koji mirne duše može da bude namenjen jednom pravom serveru, numerička sposobnosti pogodna za CAD/CAM i, najzad, pune komunikacione mogućnosti sa modemom i faks mašinom.

Kvalitet izrade je na nivou najboljih koje smo do sada videli kod nas. Velika Tower kutija je veoma solidno urađena, sa zaista velikim izvorom napajanja od 350 VA, što bi trebalo da zadovolji i najambicioznije korisnike.

Ovakva konfiguracija nas je stavila na velike muke: sa čim je porediti? Toliko je jaka u svim oblastima da bi se sa pravom mogla porediti i sa drugim mašinama namenjenim ulozi servera, ali i sa mašinama namenjenim radu sa CAD/CAM-om, a u svim slučajevima rizikujemo da ne budemo fer. Da bismo razrešili ovu dilemu, dacemo konačnu ocenu u dva konteksta: ocenjivaćemo je kao server i kao CAD/CAM/DTP mašinu.

Kao server, bila bi, naravno, daleko jeftinija, jer niko ne bi za tu ulogu uzimao skupi veliki monitor, čime bi ukupna cena osetno pala i dozvolila da se razlika preraspodeli. Recimo, lako bi se mogao ubaciti onaj drugi SRAM čip na kontroler, što sigurno ne bi štetilo performansama. Performanse diska, sigurno najvažniji aspekt servera pored količine memorije sa kojom raspolaže, spadaju među dobre, ali ne i izuzetne. U neku ruku, ovo je i razumljivo, jer je danas sve veći broj IDE diskova sa daleko većim kešom na sebi – primera radi, neki stariji modeli firme „Quantum" su još onda imali po 64 KB keša, a danas se već ide na 96, pa i 256 KB. Kada se zna da je mehanika diskova u osnovi ista, a da razlike potiču, pre svega, od algoritma za keširanje i količine keš memorije, ovaj ESDI disk bi sigurno radio bolje sa još toliko memorije na inače veoma ubedljivom kontroleru. Ovako, disk je pokazao dobre performanse, ali već je previše IDE diskova koji imaju iste, pa i bolje performanse za ista ili manje para – a ni kapacitet nije više problem, sada se masovno nude IDE diskovi od 300, 420 i 510 MB (Quantum, Fujitsu, Conner).

Što se ostalih osobina sistema tiče, mirne duše bi mogao biti server u manjim mrežama.

Kao CAD/CAM/DTP stanica, ovaj sistem ima veći disk nego što bi bilo nužno – za takve poslove obično je 210 MB sasvim dovoljno. Monitor i video karte su aduti na koje se u tom slučaju igra, a u toj igri ovaj sistem je spreman da odbrani svoje pozicije protiv svih, ili po osnovu apsolutnih mogućnosti, ili po osnovu cene, ili oboje. Jedinu senku sumnje baca odnos kvaliteta i cena u poređenju sa matičnom pločom sa 80486 na 33 MHz, koja se po ceni verovatno jako približila ceni ove ploče sa koprocesorom, a po performansama je ipak oko dva puta brža. No, čak i uz izmenu platforme, ne bismo menjali ni video kartu, ni monitor (uzgred, ovo je prvi veliki monitor koji nam se zaista dopao, do sada samo uvek bili srećni da se vratimo na naš mali 14-inčni Mitac AL-4020).

Na kraju, za sve one koje zanima fax/modem kartica, a takvih je izgleda dosta, u „Mikronovi" su nam rekli da oni već duže vreme prodaju ovakve uređaje i kao samostalne artikle i da ih imaju na lageru. Prema tome, ko voli, nek izvoli.

Ovo prvo pojavljivanje „Mikronove" u „Računarima" je zaista sjajno; nadamo se da nije i poslednje, kao i da je sva oprema koju prodaju na ovom nivou. Ako je tako, onda ostali prodavci, uključujući i najpoznatija i najcenjenija imena, treba dobro da se pripaze.

Korisna adresa

Micronova
21000 Novi Sad, Lenjinov Trg 5
Tel. 021-21 821, Fax 021 28 042

Nastavak sa strane 21

ogradu – nismo probali rad pod *AutoCAD*-om, za koji Diamond kaže da poseduje veoma brze pobudne programe. S druge strane, proizvođač i prodavac zaista nemaju čega da se stide, jer su dobijeni rezultati u samom vrhu današnje tehnologije u ovoj klasi, što važi za sva tri proizvoda. Zato i ne treba očekivati velike razlike.

MRTVA TRKA

Realno gledano, radi se o tri kartice manje-više iste brzine. One potiču iz poznatih i cenjenih izvora, pa ni tu nema bitnije razlike. Kvalitet podrške u trenutku kupovine je opet približno sličan u slučaju ATI i „Tseng" kartica, a odskače u slučaju „Diamond" kartice.

U poređenju sa konkurencijom, ATI kartica nudi osam puta više boja za istina spolja unesena daleko, ali to obuhvata i, recimo, slike skenirane sa HP ScanJet IIc, o kome je bilo reči (TIF format sa 16,7 miliona boja). „Tseng" kartica to ne može. Drugo, ATI kartica nudi i pomalo začuđujuće kvalitetnog miša, prdé vezanog na nju a ne na serijsku vezu, što je određena prednost, pod uslovom da se na odrazi bitnije na cenu. Sa svoje strane, Diamond kartica nudi odličan dopunski program za obradu slika u dubokoj boji, što je konzistentnije i, rekli bismo, potrebnije u slučaju video karte no miš (mada se ni bez njega danas više ne može).

Najzad, pitanje cene. Očekuje se da će cena ATI kartice biti otprilike koliko i Tseng kartice sada, +/- 5%. Ako tako i bude, onda će ATI kartica biti bolji posao, samo zato što za iste pare daje više, miša sigurno, a ako ste profesionalno orijentisani ka izdavačkoj delatnosti, i veći broj boja je nesumnjiva prednost.

U međuvremenu, „Tseng" karticu možete dobiti odmah, na jednostavan telefonski poziv. Pošto se radi o daljem razvoju dobro poznatog i u praksi osvedočenog modela, teško da sa njom možete pogrešiti, mada objektivno ne nudi bilo šta sem golog hardvera i veznika.

„Diamond Speedstar VGA" je posebna priča. Imamo utisak kao da se proizvođači spremni da je iz popularne prebace u poluprofesionalnu klasu. Njena vrlina nije toliko brzina koliko celovitost paketa u kome se prodaje: glavni akcenat razvoja veznih programa je bačen na *Windows* i *AutoCAD*, naravno sa dubokom bojom.

Stavite bilo koju od ovih kartica u neki skromniji AT ili SX i uživaćete u brzini prikaza – izgledaće kao da ste zamenili matičnu ploču. Zadovoljstvo garantovano. Nećete pogrešiti ni sa jednim od ovih proizvoda u celini, i uz uslov da ne košta osetno više od konkurencije, mi bismo se opredelili za „Diamond Speedstar" karticu samo zato što je kao proizvod najzrelija i najcelovitija od sva tri konkurenta.

Korisne adrese

Mikronova
Lenjinov Trg 5, 21000 Novi Sad
Tel. 021/21-821, 611-671, fax 021/28-042

Geme Electronics
Beograd, Mišarska 11
Tel. 011/332-275, 339-494, fax 011/335-902

Micro Air
Beograd, Save Kovačevića 23
Tel. 011/458-948, fax 011/453-352

UZDUŽ I POPREKO

Neka istraživanja (USA) govore da su programi za unakrsna izračunavanja (*spreadsheet*), pored tekst procesora, u samom vrhu po broju primena. Namera nam je da u ovom članku napravimo poređenje između nekoliko najpopularnijih *spreadsheet* programa, kao i da vas informišemo o nakim novitetima na tој sceni.

Vladimir Stamenović

Spreadsheet programi zauzimaju jedan veliki segment tržišta softvera za PC računare. Neki ih nazivaju i tabelama, radnim tabelama, tabelama za unakrsno izračunavanje ili elektronskim tabelama. Kako god ih mi budemo zvali, oni su postali sastavni deo života ogromnog broja korisnika PC-ja.

Prvi *spreadsheet* program, *Visi Calc*, urađen je u pre IBM/PC era za „Apple II". Od tada do danas zna mo za klasične kolone i redove. Prelomni trenutak se desio nakon pojave IBM/PC računara, negde 1985. godine, kada je „Lotus" napravio čuveni *1-2-3*. Kažu da se tada računar kupovao da bi se radilo sa tim programom. Ukratko, ono što je dovelo do ovolike popularnosti tabelarnih programa je relativno jednostavan način za računanje, planiranje, pregled, izmene kao i štampanje raznih izveštaja na osnovu podataka sa kojima se radi. Srce svih *spreadsheet* programa je magična tabela koja se automatski prilagođava svakoj promeni podataka. Ako promenite samo jedan broj, svi podaci koji su vezani za njega će se prilagoditi novoj vrednosti u sekundi.

Tokom vremena ovi programi su evoluirali, a te zultati su i dobri i loši. U dobre strane razvoja *spreadsheet* programa ubrojao bih fantastična unapređenja vezane za grafičku prezentaciju podataka kao nova načine za pregled i manipulaciju podacima. Te dobre strane su izazvale one loše, koje se ogledaju u povećanim zahtevima za dobrim hardverom. Primera radi, *Windows* verzija *Lotus 1-2-3*, najprodavanijeg *spreadsheet*-a na svetu (preko 15 000 000 registrovanih korisnika), zahteva više od 8 MB na disku, a 386 PC ima dosta probleme da zadovoljavajućom brzinom obavlja posao. Slična situacija je i sa programima drugih proizvođača.

U ovom momentu raspolažemo sa dva „Lotusove" proizvoda: *1-2-3* verzije 2.3 i 3.1+, „Borlandovim" *Quattro Pro* verzija 3.0 i „Microsoftovim" *Excel* verzija 3.0 (ramo i *1-2-3* za *Windows* ali u beta verziji, pa smatramo da nije uputno da nekompletan program učestvuje u testu. U svakom slučaju, saopštićemo vam svoje utiske o njemu. Isto tako, dačemo vam i neke interesantne informacije o *Lotus Improve* *spreadsheet* programu za NeXT kompjutera i to se mo kao ezoteriju u svetu ovakvih programa.

Navedeni programi čine oko 95% tržišta softvera ovog tipa, pa mislim da ćete biti u mogućnosti da se definitivno opredelite. Pri tome sam pošao od činjenice da se *spreadsheet* programi najviše razlikuju u lakoći upotrebe, brzini rada, broju ugrađenih funkci ja i komandi, kao i u mogućnostima za grafičku prezentaciju podataka. Još nešto je vrlo bitno. U ovom poslu smatram (i ne samo ja) da je *Lotus 1-2-3* standard, pa ću praktično sve ostale porediti sa njim. Neće to biti baš lako, jer na primer *MS Excel* je pravi *Windows* program, dok svi ostali to nisu, ali ću se potruditi da budem maksimalno objektivan.

Kao prvo, evo kratkog prikaza svakog od navedenih programa.

QUATTRO PRO 3.0 (DOS)

Ovo što je firma „Borland" napravila svojim mogućnostima prevazilazi osnovnu namenu *spreadsheet* programa. U njemu se koristi GUI (*Graphical User Interface*) a sve je u WYSIWYG stilu. Ja bih ga pre nazvao integrisanim paketom na bazi *spreadsheet*-a, jer su dodele vrlo impresivne grafičke mogućnosti obrade tabela, grafikona i slika. Zahvaljujući originalnoj VROOMM tehnologiji, odnosno dinamičkom korišćenju memorije, *Quattro Pro* može da radi i na XT računarima.

Prvo što pada u oči nakon startovanja programa je veoma lepo urađen interfejs. Komandni meni je padajući odnosno u *pull-down* stilu. Deluje i funkcioniše uglavnom dobro. Uglavnom, jer se dešavalo da se neki od delova menija „zamrzne" i ostane na ekranu tako da ga ne možete obrisati! Ukoliko ste korisnik *Lotusa 1-2-3* pa vam ovakav meni ne odgovara, možete ga podesiti da bude u „Lotus" stilu, odnosno u jednoj liniji iznad tabele. Komande su vrlo logično raspoređene po meniju. Osim komandnog menija, tu je i meni za rad sa mišem, kao i nekoliko ikona.

Quattro Pro ne može da radi sa više tabela u okviru jedne datoteke. Umesto toga, možete otvoriti nove prozore (32 maksimalno) u kojima će biti različite tabele. Prozori mogu biti proizvoljnih veličina i na različitim mestima na ekranu. Zgodno je što se mogu linkovati podaci između različitih tabela. Pri tome tabela ne mora da bude otvorena u memoriji da bismo koristili mogućnost linkovanja. To možemo ostvariti i sa datotekom na disku.

Za **Edit** komande moram da kažem da se izvršavaju dosta sporo. Čak i u malim tabelama treba čekati nekoliko sekundi da se izvrši, recimo, komanda **Copy**. Prvo što smeta je skrolovanje ekrana koje se javlja pri izvršavanju komandi i to po dva do tri puta. Sistem je otprilike ovakav: izdate komandu, na primer **Insert Rows**, ekran skroluje, pojavi se indikator **Wait**, *Quattro Pro* ubaci praznu kolonu u tabelu, ekran skroluje, indikator kaže **Ready**, ali odmah zatim BKGD (rekalkulacija u pozadini), pa ponovo **Wait** i tek nakon toga konačno **Ready**. Sve to traje dosta dugo.

Prave vrednosti GUI ćete osetiti i videti tek kroz komandu **Style**. Ona u jednom delu sadrži standardne opcije za određivanje formata u kojem će biti prikazani brojevi, širine i visine ćelija i zaštite podataka u ćelijama. Drugi deo **Style** menija sadrži opcije za kvalitetnu prezentaciju podataka. Tu možete crtati linije ili bokseve oko ćelija, menjati tipove i veličine slova, senčiti ćelije... Na žalost, i ovde važi primedba o sporosti izvršavanja komandi.

Po mom mišljenju, najkvalitetniji deo programa je onaj koji se odnosi na grafičku prezentaciju podataka. Ovo što nudi *Quattro Pro* ne može se naći ni u jednom drugom *spreadsheet* programu za DOS. Praktično su ujedinjene klasične mogućnosti kreiranja grafika na bazi podataka iz tabele i, u manjoj meri, dodatne opcije nekih specijalizovanih programa za grafičku prezentaciju podataka. Postoji veliki broj mogućnosti ulepšavanja i poboljšanja izgleda grafikona. Naime, moguća je promena boja svih elemenata u grafiku, promena oznaka (punjenja) elemenata grafikona, pa čak i primena različitih tipova grafikona za pojedine serije podataka. *Quattro Pro* nudi ukupno 14 tipova grafikona, od kojih su četiri 3D grafici. *Quattro Pro* ima sve što treba da vaše podatke grafički prikažete na najbolji mogući način.

Posebna priča i dobra igračka je *Graph Anotator*. On omogućuje dodatno poboljšanje grafika kroz mogućnosti promene skoro svih elemenata grafikona i dodavanja novih. *Anotator*-ov interfejs je izuzetan ne samo po izgledu već i po funkcionalnosti. U *Anotator*-u možete menjati boje, šrafure, tipove slova, poziciju



Snimio: Nenad Marjanović

Izvanredan u grafičkoj prezentaciji podataka: *Quattro Pro*



Quattro Pro: izgled ekrana i štampanog dokumenta

elemeneta na grafikonu ili dodavati neke nove kao što su linije, strelice, tekst, gotove sličica (*clipart*) itd. itd. Odlično urađeno, tako za rad, a sa malo truda i mašte napravićete grafikon – sliku profesionalnog izgleda. Posebno ističem mogućnost ubacivanja „grafičkog tastera" u grafikon – sliku. Taj taster aktivira neku vašu makro proceduru ili poziva drugi grafikon! Interesantna je mogućnost za linkovanje elemenata grafikona sa podacima iz tabele, kreiranje taksi grafikona, pravljenje sopstvenih slika ili znakova i, kao najinteresantnija, dodavanje zvučnih efekata grafiku. *Quattro Pro* nudi jedinstvenu mogućnost da prezentaciju podataka propratite zvučnim efektima snimljenim u SND datotekama!

Kada napravite zadovoljavajuće grafikone i slika možete ih povezati u **Slide Show**. Brojni su načini međusobnog smenjivanje slika tokom prezentacije, a trajanje zastoja pojedine slike na displeju vi određujete. Naravno, sve to možete začiniti zvučnim efektima melodijama. Moram priznati da takva mogućnosti nisam očekivao od ovakvog tipa programa. Čestitke proizvođaču.

Quattro Pro nudi dosta „desktop publishing" tačnije „spreadsheet publishing" mogućnosti. Ne morate imati laserski štampač da biste dobili lepo odštampan izveštaj. U tu svrhu na raspolaganju imate par korisnih alata. Oni omogućuju štampanja uzduž papira tako da dugačke, odnosno široke tabele možete štampati na neprekidnom papiru u jednom delu. Dalje, možete štampati tabelu zajedno sa umetnutim grafikonom ili odabrati opciju koja redukuje veličinu tabele tako da stane na jednu širinu ili na najmanji mogući broj strana, a da pri tome veličina slova omogućuje normalno čitanje podataka. Ovo je posebno korisno pri radu sa laserskim štampačima. Pri štampanju ima samo jedna sitnica koja mi je smetala. Naime, ne postoji mogućnost štampanja u „pozadini", tako da u toku štampanje ne možete raditi ništa.

Kao i većina *spreadsheet* programa, *Quattro Pro* nudi mogućnost rada sa bazom podataka. To podrazumeva sortiranje podataka, po maksimalno pet ključeva, kao i pretraživanje baza po zadatim kriterijumima. Ako ste korisnik *Paradox*, *Reflex* ili *dBASE* programa za baze podataka, onda možete iz *Quattro Pro* ostvariti direktan pristup podacima bez prethodnog prevođenja. Baza je ograničena brojem redova u tabeli, tako da možete imati najviše 8192 zapisa. U maloj tabeli, oko 20 K, sortiranja baza podataka koja se sastoji od 87 zapisa sa po 6 polja, a po 5 ključeva trajalo je 23 sekunde, sa par neizbežnih skrolovanja ekrana.

Spreadsheet programi su idealni za statistiku i analizu podataka. *Quattro Pro* nudi rad sa matricama, regresionu analizu, pa i specificiranje linearnih modela u tabeli u cilju minimiziranja ili maksimiziranja nekih kombinacija varijabli u vašem modelu. Interesantna je mogućnost preračunavanje formule unazad. Tu startujete sa određenim rezultatom, a tražite izračunavate varijable koje proizvode taj rezultat.

Korisnici *Quattro Pro* imaju mogućnost da, kroz makro komande i procedure, znatno unaprede svoj rad, bilo da je reč o jednostavnom ubrzanju rada u tabeli ili o kreiranju kompleksnih aplikacija. Za one koji rade sa *Lotusom 1-2-3*, značajno je da *Quattro Pro* podržava sve makro komande do verzije *2.01*. Procedura možete čuvati u biblioteci makroa, što je najbolji način, i koristiti ih u bilo kojoj tabeli. *Quattro Pro* sadrži više makro komandi od drugih *spreadsheet* programa, što je od velike pomoći u radu. Veoma je značajno to što *Quattro Pro* ima *Macro Debugger* iz koga se program može izvršavati korak po korak ili između određenih tačaka u programu (*breakpoints*), tako da je olakšano pronalaženje i lociranja grešaka. „Borland" je učinio maksimum da bi zadovoljio potrebe programera u *Quattro Pro*.

Quattro Pro je dobar *spreadsheet* program, čak i više od toga, obzirom na velike mogućnosti u grafičkoj prezentaciji podataka. Programeri će biti zadovoljni jer *Quattro Pro* ima moćne i brojne makro komande koje, u sprezi sa gomilom @ funkcija i dibagerom, omogućuju kreiranje odličnih aplikacija.

Quattro Pro ima svojih mana, verovatno ste to primetili kroz tekst. Mogao bih da ih svedem na nekoliko grupa. Pre svega, jako mi smeta sporost u izvršavanju komandi. Činjenica je da i kod drugih programa sa GUI ima više ili manje, a uglavnom manje, skrolovanja ekrana, kao i da se neke komande sporije izvršavaju. Međutim, *Quattro Pro* za sta zamara i odbija svojom sporošću. To se najviše uočava kod **Edit** i **Style** komandi, mada ni neke druge (**Sort**, **Debug**) nisu sjajne. I pored velikog truda da se korisnicima omogući maksimalni komfor kod štampanja, mnogo nedostaje mogućnost štampanje u pozadini, odnosno mogućnost da se nastavi rad za vreme štampanja. Na kraju, čini mi se da su, zbog obilja dodatih mogućnosti, neki delovi menija postali isuviše komplikovani što bi, i pored logičnog rasporeda komandi, moglo izazvati konfuziju kod neiskusnih korisnika.

Sve u svemu, *Quattro Pro 3.0* je dobar softver prijateljski nastrojen ka iskusnijim korisnicima *spreadsheet* programa, sa izuzetno dobrim delom za grafičku prezentaciju podataka, kao i mogućnostima za programiranja. U našim uslovima važno je i što nije mnogo zahtevan u pogledu hardvera. Početnici će morati dobro da zasuču rukave dok ga ne savladaju.

LOTUS 1-2-3 2.3 (DOS)

Lotus 1-2-3 verzija *2.3* radi na IBM PC/XT i AT računarima sa najmanje 384 K RAM-a. Ako ćete raditi i sa WYSIWYG-om, onda je minimum 512 K. Ovaj program i *Quattro Pro* su u direktnoj konkurenciji i bore se za onaj deo tržišta *spreadsheet* programa u koma korisnici ne žele veća izdvajanja za unapređenje svog hardvera.

Korisnički interfejs nije značajnije izmenjen u odnosu na prethodne verzije. Međutim, sa WYSIWYG-om izgleda prilično atraktivno. I dalje postoji čuveni „Lotusov" meni iznad tabele sa poznatim rasporedom komandi. Sa strana se nalaze ikone za pomeranje pointera i pozivanje helpa pomoću miša. Rad sa mišem je znatno unapređen i lakši je nego u *Quattro Pro* ili *1-2-3* ver. *3.1*. Inače, sva podešavanja idu preko dijalog boksova koji sadrže veliki broj mogućnosti koje se mogu upotrebiti, a smeštene su na jednom ekranu. Promenu u podešavanjima možete vršiti i preko tastature i mišem, s tim da je rad sa mišem dosta zgodniji i brži. **Help** je znatno unapređen u odnosu na sve prethodne verzije, računajući i *3.1*. Izuzetno je „kontekstno senzitivan".

Uz *1-2-3* ide i nekoliko dodatnih programa koji inače dobar program čine znatno boljim, lakšim i korisnijim. Tu je WYSIWYG, koji omogućuje izuzetno kvalitetnu prezentaciju podataka iz tabele. *Auditor* kao dodatak programu pomoću koga se vrlo lako prate međusobne zavisnosti formula i otkrivaju greške. *Viewer* omogućuje pregled tabela pre učitavanja, izuzetno jednostavno linkovanje ćelija iz tekuće i neke druge tabela na disku, kao i još neke pogodnosti. Tu su i *Macromanager* i D360, koje smo već videli u prethodnoj verziji. Treba još napomenuti da se uz ovaj paket isporučuje vrlo kvalitetan program za učenje *1-2-3*, smeštan u poddirektorijum TUTOR, a nazvan *1-2-3-Go!*, kao i program za obuku za rad sa WYSIWYG-om koji se zove *Wysiwyg-Go!*.

WYSIWYG „spreadsheet – publishing" dodatni program predstavlja značajno unapređenje u odnosu na prethodni *Always*. Prvi put je viđen u *1-2-3* ver. 3.1. Time su se smanjile razlika među verzijama *1-2-3* za „sve" i *1-2-3* za IBM PC/AT računare. WYSIWYG omogućuje kreiranje fantastičnih tabela, što sa izgleda liče, i to potpuno jednakog na ekranu i štampaču.

LOTUS 1-2-3 ZA WINDOWS

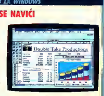
TREBA SE NAVIĆI

Ovaj program nije uključen u test, jer ga imam samo u beta verziji, mada je već nekoliko meseci u prodaji. Obzirom da su takvi programi uglavnom nekompletni i da se dosta razlikuju od onih koji se pojave na tržištu, daću vam samo neke osnovna informacije o njemu.

Mada je „Lotus" među prvima napravio *spreadsheet* sa pravim grafičkim interfejsom, *123/G* za OS/2, sa verzijom za *Windows* (*123/W*) se pojavio sa malim zakašnjenjem. Proizvodnja ovog programa je produžena jer se nastojalo da to bude potpuni *Windows* program, ali i da zadrži 100% kompatibilnost sa drugim verzijama. Zato možda i ima kritičara kojima takav pristup smeta, jer je „Lotus" morao da pravi neku vrstu kompromisa. Uglavnom, u *123/W* možete učitati sve postojeće datoteke, koristiti već napisane makro procedure i raditi sa poznatim kombinacijama komandi iz prethodnih verzija.

Da bi se olakšao rad, uveden je *Smarticon* sistem ikona preko kojih možete odmah aktivirati niz komandi, makroe ili grafike. Te ikone možete menjati i kreirati po volji, povećavati i smanjivati, kao i pozicionirati bilo gde u tabeli. Posao možete obavljati i mišem i preko tastature. Svejedno je da li ćete prvo izdati komandu, pa onda odabrati oblast ćelija ili obrnuto. Osim uobičajenog padajućeg menija karakterističnog za *Windows* aplikacije, paralelno postoji i meni u „Lotus" stilu sa svim poznatim komandama.

Inače *123/W* podržava sve prednosti *Windows*-a, kao što je puni WYSIWYG, formatiranje teksta, izgledo strana za štampu, padajuće menije, dijalog boksova, potpunu kontrolu rada preko miša, direktno određivanje širina i visina kolone itd. Moguć je istovremeni rad sa više dokumenata u memoriji (MDI) u različitim prozorima, koji se mogu proizvoljno dimenzionirati i pozicionirati. *123/W* podržava i DDE, odnosno razmenu podataka među aplikacijama. To takođe znači da možete povezati *123/W* aplikaciju sa nekom drugom *Windows* aplikacijom koja podržava DDE. Osim linkovanja, zadržana je jedinstvena mogućnost rada u trodimenzionalnoj tabeli, što svakako pozdravljamo.



Ni oni nisu odoleli: *Lotus 1-2-3 for Windows*

Direktan pristup eksternim bazama podataka (*dBASE III*, *IV*, *Paradox* i *SQL Server*) je omogućen kroz *DataLens* tehnologiju. Ugrađen je i *Solver* koji postoji i u *Excel*-u.

Kao i do sada, „Lotus" je predvideo i alate za pravljenja aplikacija, kako za korisnike, tako i za one koji razvijaju aplikacije za druge. Znatno je proširen spisak makro komandi, a već je u prodaji *Lotus Add In Developers Kit* (ADK), pomoću koga možete pisati i kompajlirati aplikacije za druge korisnika *123/W*. Interesantno je da ADK dozvoljava direktnu integraciju C programa u *123/W*.

Izveštaji pravljeni sa *1-2-3* i *WYSIWYG*-om, po kvalitetu mogu se meriti sa onima rađenim u specijalizovanim grafičkim i desktop publishing programima. Među pogodnostima koje možete koristiti spadaju i crtanje okvira ili senčenje ćelija ili oblasti ćelija, promena tipa, stila ili veličina slova, ubacivanje grafikona u tabelu ili maksimalno doterivanje izgleda grafikona. *WYSIWYG* sadrži četiri Bitstream tipa fontova *Courier, Swiss, Dutch* i *Xsymbol* a korisnik može birati veličinu od 3 do 72 tačke u kombinaciji sa bold, italic i podvlačenjem. Moguća je upotreba 8 različitih fontova istovremeno, s tim da se mogu napraviti mnoge različite varijante koje ćete čuvati u posebnoj biblioteci fontova i pozivati po potrebi. Veoma je interesantna i upotrebljiva mogućnost da se sačuva neki format pod određenim imenom. Na primer, ako često koristite *Swiss* tip fontova u veličini od 14 tačaka, na tamno plavoj pozadini sa žutim tekstom, onda možete koristeci komandu **Named-Style** toj kombinaciji dati ime i tako ćete je pridružiti bilo kojoj ćeliji ili oblasti kada zatreba. Brojne su druge mogućnosti koje možete koristiti: promena boja pozadine, pointera, negativnih brojeva, ćelije sa skinutom zaštitom, okvira, izbor broja redova koji se vide na ekranu, pa čak i jednostavni tekst procesor. Osim toga, može se ubaciti i grafikon (PIC fajl) ili neka slika (CGM metafajl) u tabelu. I ne samo to – vrlo su velike mogućnosti doterivanja i ispravki slike i grafikona kroz meni koji dosta sličan onome u *Lotus Freelance* grafičkom paketu. Grafik se može poboljšati, na primer, dodavanjem teksta, linija, strelica, pravougaonih ili kružnih-elipsastih elemenata ili simbola. Tu su i mogućnosti pomeranja, zakretanje ili rotacija, kao i promena veličine grafika, slika. Što se tiče boja, na raspolaganju su 224 nijanse koje možete izabrati kroz meni.

Sledeći dodatni program koji možemo koristiti u *1-2-3* je *Auditor*. Isti je i u verziji *2.3* i u verziji *3.1+*. On ima svoj sopstvani meni u „Lotusovom" stilu, kroz koji je vrlo jednostavno naći poziciju svih formula u tabeli, sve formule u tabeli koje zavise jedna od druge, pa i redosled kojim preuzimaju podatke jedna od druge. Osim toga, lako ćete naći onu ili one ćelije koje su svojim sadržajem uspostavile „kružni tok" formula.

Veoma koristan dodatni program je *Viewer*. Bez napuštanja *1-2-3* startovanjem *Viewer*-a možete pregledati sadržaj svih WK1, WKS i WRK datoteka i zatim onu koju želite učitati u *1-2-3*. U *1-2-3* je predviđena mogućnost da u aktivnoj tabeli koristite podatke iz tabela sa diska, dakle da povezujete više tabela u jednoj. *Viewer* čini taj posao lakim, pa i zabavnim: jer automatski ubacuje vezu i adresu odabrane ćelije sa diska u aktivnu tabelu.

Kroz *WYSIWYG* ima dosta poboljšanja u radu sa grafikonima i grafičkom prezentacijom podataka. Osim toga, tu su i dva nova tipa grafikona: HLCO (**high-low-close-open**) i **Mixed** (mešani). Obe ova tipa smo videli u *1-2-3* verzije *3.1* i *3.1+*, kao i u *Symphony*. Moguće je prikazivanje bar grafika horizontalno ili vertikalno, kao i kvazi 3D grafika. Ni ova verzija popularnog programa nema pravu 3D grafičku prezentaciju podataka.

Postoje mogućnosti štampanja **Encoded** datoteka, kao i štampanja „u pozadini". Ova poslednja je izuzetno korisna, jer omogućuje nastavak rada u tabeli dok printer štampa. Ono što mi tu smeta je da se to mora predvideti pre startovanja *1-2-3*. Naime, ako hoćete tu mogućnost, morate prvo pokrenuti program *BPRINT*, pa tek onda *1-2-3*, u okviru koga treba odabrati **Background** opciju iz **Print** menija. Povoljno je to što *BPRINT* zauzima samo 6 K RAM-a, a ako vam



Jedinstveno svojstvo Lotus-a: Višedimenzionalne tabele u verziji 3.1

i to zatreba, moraćete da resetujete računar. Dakle, reč je o veoma korisnoj mogućnosti koja je mogla biti malo spretnije uređena. Za štampanje grafikona zadržana je stara koncepcija koja predviđa poseban program *PrintGraph* koji služi isključivo za štampanje grafikona. Čak je i dizajn nepromenjen u odnosu na stare verzije. Čudno je da „Lotus" nije našao način da omogući štampanje grafikona iz jednog menija, dakle bez *PrintGraph* programa.

Sa ovim paketom radio sam na četiri različita računara. Prvo o IBM PC/XT-u: *1-2-3* stvarno lepo radi na XT-u sa 512 K RAM-a. Brzina je bila u skladu sa hardverom, ali je bitno da nije bilo problema sa

LOTUS IMPROV

EZOTERIJA U SVETU UNAKRSNIH TABELA

Svi *spreadsheet* programi u principu liče jedan na drugog i izgledaju i rade kao ili slično „Lotusu" *1-2-3*. Poboljšanja koja donose nove verzije o kojima smo govorili, ipak ostaju samo to, odnosno ne donose nikakva nova, revolucionarna unapređenja ili ideje.

„Lotus" opet je korak dalje. Krajem 1990. godine predstavljen je potpuno novi *spreadsheet* program – *IMPROV*. Skoro da nema nikakve sličnosti sa *1-2-3*, a neki elementi interfejsa nisu ni sada, ni do sada viđeni u nekom drugom programu. Dakle, reč je o nečemu stvarno novom. Nisam imao prilike da radim sa ovim programom, ali mislim da će vam biti zanimljivo da vidite u kom pravcu će se najverovatnije kretati dalji razvoj *spreadsheet* programa.

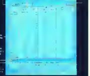
Kažu da je *Improv* teže opisati nego raditi sa njim. Naime, svaka sličnost sa današnjim *spreadsheet* programima prestaje posle činjenice da oni imaju redove i kolone. Nema više kolona A, B, C... i redova 1, 2, 3... Sada se njima mogu davati imena po želji i ona predstavljaju identifikatore koje „Lotus" zove **Items**. Lokacija ćelije u tabeli je određena preko odgovarajućih **Items** (npr.: Januar Suma). Kada startujete *Improv*, dobijate tabelu sa jednom ćelijom!! Pretpostavka je da ćete kreirati, upotrebiti onoliko ćelija koliko vam treba! Ima nekoliko načina za kreiranje tabele i svi potvrđuju filozofiju *Improve*-ovog interfejsa: najlakše je uraditi one operacije koje najčešće koristite.

Improv je napravljen ekskluzivno za NeXT računare. Oni koji prate novosti u računarskoj tehnici i tehnologiji sigurno znaju da je reč o veoma dobrim i moćnim mašinama koje ne rade pod DOS-om. Obzirom da se 99% ovih računara prodaje u USA, *Improv* je još praktično nepoznat u Evropi, a nema nikakvu reklamnu podršku.

„Lotus" je planirao da napravi *Improv* i za druga okruženja, a pošto je zavisan od grafičkog interfejsa, otpnike će kandidati biti OS/2, *Windows*, *XWindows* ili makintoš".

Informacije radi, navešću vam nekoliko najinteresantnijih činjenica vezanih za ovaj program:

- Razni tipovi **Items** su predstavljeni kroz kategorije. Kategorija je kratka, obično jedna reč, kojom se opisuje tip informacija predstavljen sa **Items**. Na primer, ako red sa **Items** predstavlja mesece u godini, a kolone proizvode koje prodajete, onda možete definisati kategorije Meseci i Proizvodi. Njih možete kombinovati po želji.
- Bilo koji broj **Items** može sačinjavati grupu, čija imena se koriste svuda gde treba zvati **range**. Znači **range** i dalje postoji i označava se kao u *1-2-3* sa „..." između **Items**.
- Interfejs nije kompatibilan sa *1-2-3*. *Improv* uopšte ne liči na *1-2-3*. Pritiskom na „/" se ne pojavljuje meni. Međutim, interfejs je toliko jednostavan da kad shvatite hijerarhiju kategorija, grupa i **Items**, sve teče lako i prirodno.
- Svaka tabela može imati više pogleda zgleda gde svaki pogled predstavlja podatke na drugačiji način. Jedna *Improv* datoteka (naziva se model) može da ima više tabela, svaka sa svojim podacima. Ćelije u drugim tabelama u istom modelu mogu biti korišćene u svim formulama lako, što se napišu njihove adrese koje se sastoje od imena tabele iza koga sledi dvostruka dvotačka (naprimer tabele Godina91 i Godina92: Januar Suma Januar92 Suma92).
- Korišćenje ćelija i grupa ćelija je najčešće za izvršavanje kalkulacija na njima. Upravo tu je „Lotus" ponudio još jednu pravu inovaciju. Mi znamo da u *1-2-3* za svaku kalkulaciju nad ćelijama moramo napisati odgovarajuće formule. U toj situaciji sasvim je normalno imati desetine i stotine formula u jednoj tabeli. *Improv* je prekinuo vezu između ćelija i formula. Formule se unose u tako zvani „formula panel" polprozor u *spreadsheet* prozoru, koji se naravno može povećavati i smanjivati po želji. Svaka formula ima desnu stranu koja opisuje kalkulaciju i levu stranu koja specificira poziciju rezultata (u množini). U množini zato što jedna formula može dati više rezultata koji se mogu pozicionirati na različitim mestima u tabeli/tabelama. Na primer, ako imate **Items** Količina, Cena i Suma, onda je Suma = Količina Cena. Ova formula će izračunavati sve Sume u tabeli.
- Ubacivanje formula u tabelu retko kad zahteva upotrebu tastature. Postoji „formula panel" sa paletom matematičkih simbola koje duplim „klikom" miša ubacujemo u formulu, a klikom u tabeli dodajemo formuli identifikatore. Ukoliko napravimo grešku, odmah ćemo biti obavešteni preko poruke greške i kratkim objašnjenjem zašto je do toga došlo. Klikom miša dobićemo dodatna detalj na objašnjenja o grešci i poziciji iste u formuli.



Novi koncept u radnim tabelama: *Lotus Improv*

- Sve funkcije *1-2-3* su duplicirane u *Improv* tako da se tabele rađene u *1-2-3* mogu importovati u *Improv*, a one rađene u *Improv* se mogu eksportovati u *1-2-3*. Pri importovanju *Improv* će redukovati broj potrebnih formula, a to je naročito izraženo kod velikog broja kopiranih formula koje će se svesti na svega nekoliko.
- Stari problem oko slučajnog prepisivanja formule nekim novim podatkom ovde je uspešno rešen. *Improv* ne dozvoljava upis podataka preko formule.
- Grafika iz podataka iz tabela se pravi pomoću **Presentation Builder** alata. Možete dodati slobodno ručno crtanje, kao i PostScript i TIFF grafičke datoteke.

Oni koji su imali prilike da vide *Improv*, kažu da on prosto nokautira korisnika, jer je stvarno jedini NOV proizvod koji se pojavio nakon duže vremena. „Lotusova" deja je prava inovacija u *spreadsheet* programima. Po „Lotusovom" mišljenju korisnici *Improv*-a bi trebalo da budu napredni i iskusni programeri u *1-2-3* kojima više ne odgovaraju limiti koje ima *1-2-3*. O ceni nećemo, jer kad pomislim koliko košta najjeftiniji NeXT računar...





Verzija za napredne platforme *Lotus 1-2-3 3.1*

skrolovanjem ekrana koje me je tako bacalo u nesvest kod *Quattro Pro*-a. *1-2-3* podjednako dobro radi na desktopu sa „Hercules" karticom kao i na laptopu sa CGA monohromatskim displejom. Naravno, nedostatak memorije se brzo javlja kao ograničavajući faktor. Sa 512 K, ako koristite WYSIWYG, možete praviti relativno male tabele, a nećete moći da koristite *Viewer* i *Auditor* jer nema dovoljno RAM-a. Zato, ako već radite sa IBM PC/XT računarom sa 512 K RAM-a, preporučujem vam da dopunite RAM do 640 K, pa ćete tako moći da koristite sve pogodnosti koje nudi *1-2-3* verzija *2.3*. Instalaciju sam izvršio i na AT 286/12 računaru sa VGA grafičkom karticom. Sve je dobro funkcionisalo. *1-2-3* je radio veoma brzo, WYSIWYG takođe, *Viewer* sporo, mada brže nego na XT-u. Na starom 386/16/VGA sam bio najzadovoljniji. Sve je bilo kako treba. I brzina i količina memorije i boje.

Međutim, najbitnije od svega je da su se komande izvršavale na svim tipovima računara i vrlo brzo. Sporost koja mi je veoma smetala kod *Quattro Pro*-a u ovom programu ne postoji, te skoro da nije imalo smisla meriti vremena njihovog izvršavanja. GUI koji je napravio „Lotus" je odličan. U retkim prilikama dolazi do skrolovanja ekrana koje se odvija podnošljivom brzinom, tako da ne smeta. Učitavanje i snimanje datoteka, kao i samog programa, sa/na disk je dosta brzo, znatno brže nego kod verzija *3.x*. Unapređen je i makro jezik dodavanjem nekoliko novih makro komandi koje smo upoznali u verzijama *3.x*, a zadržana je i stoprocentna kompatibilnost sa prethodnim verzijama programa.

Ukratko, glavna prednost ove verzije *Lotusa 1-2-3* nad drugim programima ove vrste su brzina i jednostavnost u radu, kao i ogroman broj dodatnih programa na tržištu koji znatno uvećavaju njegovu moć. U nedostatke bih ubrojao tri stvari: nedostatak prave 3D grafike (kod nas je vole), štampanje grafikona iz posebnog programa i visoku cenu, bez obzira što je reč o ovako kvalitetnom proizvodu.

LOTUS 1-2-3 3.1+ (DOS)

Od trenutka kada je izbacio na tržište *1-2-3* verzija *3.0* za DOS, „Lotus" je nastavio rad na dva koloseka. Verzija *1-2-3 3.0* je radila na PC 286 i boljim računarima sa minimalno 1Mb RAM-a, a verzije *2.x*, konkretno verzija *2.3* za DOS, je namenjena slabijim računarima.

Već sam rekao da su razlike između ove dve verzije (*2.3* i *3.1+*) svedene na minimum, pa ćemo samo ukratko videti čime se diči *1-2-3* verzije *3.1+*.

U principu, *1-2-3 3.1+* je isti kao *3.0* sa dodatkom u obliku posebnih programa *Auditor*, *Viewer*, *Backsolver* i *Solver*. Prva dva su se prvo pojavila u *1-2-3* verziji *2.3* i o njima smo govorili. *Backsolver* je dodatak koji omogućuje da se neki problem rešava unazad. Odnosno, polazi se od željenog rezultata, a ispituju se izrazi varijable koje na njega utiču. *Solver* je dodatni program koji se prvi put pojavio u *1-2-3/G* za OS/2, a sada ga ima u *1-2-3* za *Windows* i *1-2-3* za „Macintosh". On predstavlja alat za rešavanje linearnih problema, tako što polazi od promene varijabli da bi se dobio najbolji rezultat. To je znatno unapređenje u odnosu na klasične „What if" analize, a mi smo u prethodnim brojevima „Računara" pisali o tome.

Svi ovi dodatni programi su veoma zgodni i korisni i kada se upotrebe uz standardne mogućnosti *1-2-3* omogućuju prilično komforan rad i izradu dobrih aplikacija. Možda je nedostatak u tome što su to ipak dodatni programi koji se pozivaju kroz *Add-in Manager*, dakle nisu sastavni deo samog programa, a uz to svaki od njih ima i svoj sopstveni meni. Međutim, ne koriste svi sve dodatne programe, neko ih ne koristi uopšte jer im je dovoljno ono što ima sam *1-2-3*. U takvim slučajevima je to prednost, a ne mana i, po mom mišljenju, takva koncepcija je dovela do toga da je *1-2-3* u svim verzijama ubedljivo najjednostavniji za upotrebu. Navedenim dodatnim programima treba dodati i WYSIWYG. On je već opisan u delu o *1-2-3* verzije *2.3*. U ovoj verziji je to indentičan program, što pojednostavljuje rad i smanjuje razliku između različitih verzija *1-2-3*.

Glavni kvalitet ove verzije *1-2-3*, po mojoj oceni, je trodimenzionalna tabela, jedinstvena u ovom tipu softvera. Naime, i pored toga što se podaci iz tabele mogu linkovati sa podacima iz drugih tabela u memoriji ili na disku, možete da u jednoj datoteci otvorite maksimalno 256 tabela!!! Ova jedinstvena mogućnost pomaže u rešavanju mnogih problema. Tako možete podeliti podatke po tabelama, na primer za svaki mesec posebne tabele. Makro procedure možete smestiti u zasebnu tabelu i tako se rešiti brige oko mogućnosti njihovog brisanja. Podatke između različitih tabela možete jednostavno povezivati i koristiti gde god vam je volja. Treba istaći i da je rad sa više tabela komforniji i brži od rada sa više tabela koje su međusobno linkovane. Primera radi, imam aplikaciju rađenu u *1-2-3 3.1* sa 11 tabela, od oko 500 K, koja predstavlja bazu podataka sa pogonskim događajima u termoelektrani za proteklih nekoliko godina. Pretraživanje te baze po bilo kom/kojim predviđenim kriterijumima traje par sekundi, dok u slučaju kada se radi sa više međusobno povezanih zasebnih tabela to traje višestruko duže.

Ima još dosta pogodnosti koje proizilaze iz trodimenzionalne tabele, a prema nekom istraživanjima korisnici (studenti na Univerzitetu Harvard) smatraju da je to najbolje rešenje, znatno bolje nego u drugim *spreadsheet* programima. U svim ostalim osobinama i kvalitetima ovaj program je isti kao i sve ostale verzije *1-2-3*, mada se meni čini da je sporiji od *1-2-3* verzije *2.3*. No, proverićemo to u testu.

MICROSOFT EXCEL 3.0 (Windows)

Ovo je potpuno drugačiji program od svih ostalih. Napravio ga je proizvođač *Windows*-a, pa je normalno da to bude pravi i najbolji *spreadsheet* program za *Windows*-e. Istovremeno, *Excel* je najprodavaniji *spreadsheet* za ovu platformu, a isto tako i za „Macintosh".

"Pojavom *Excel*-a učinjen je veliki korak napred u razvoju *spreadsheet* programa, jer je doneo veliki broj značajnih novina: totalni grafički koncept i povezanu sa tim jednostavnost u manipulaciji grafikonima, rad sa proizvoljnim brojem prozora, linkovanje više radnih tabela, „Preview" režim rada pre štampanja, punu UNDO funkciju i izuzetne mogućnosti programiranja sa makroima. Sve su to bile potpune novosti za ostale konkurentske pakete i nije ni čudo da je *Excel* bio proglašen za najbolji softverski proizvod u 1988. godini." (M.L. Računari 71/20). Potpuno se slažem sa ovom ocenom, s tim da je *Excel*, ponavljam, *Windows* program. Da vidimo šta nam nudi.

Instalacija teče kroz *Windows*-e a program zauzima oko 6 M na disku. Zanimljivo je da se uz program isporučuje i specijalno uputstvo za upotrebu namenjeno korisnicima „Lotusovih" *spreadsheet*-ova. Oni mogu da počnu sa upotrebom *Excel*-a odmah jer on čita i snima WK? tabele a omogućuje i linkovanje sa postojećim tabelama u *1 2 3* formatu. Postoji i dobar help za takve korisnike koji precizno objašnjava razliku između komandi. Za makroe nema problema jer je je ugrađen *Macro Translator* prevodilac makro procedura iz *1 2-3* u *Excel*. Sve ovo je urađeno jer je kao što sam rekao ipak *1 2-3* standard za ovu vrstu programa.

Nakon startovanja programa, na ekranu vidite praznu tabelu sa tipičnim *Windows* menijem u prvoj liniji i *toolbar* u drugoj liniji ekrana. *Toolbar* je novitet u *Excel*-u 3.0. Slične stvar već postoji u *Ami Professional* i *Corel Draw* programima a sada i u većini svih *Windows* programa. On omogućuje da se ubrzaju mnoge radnje vezane za različita podešavanja u tabeli. Potrebno je samo da prvo označite oblast ćelija (najlakše mišem) u kojoj nešto radite pa onda da aktivirate odgovarajuću ikonu-taster iz toolbar-a. Prema istraživanjima „Microsofta" oko 70% svih operacija u tabelama se svodi na sumiranje vrednosti po redovima ili kolonama. Jedna od *Excel*-ovih originalnih mogućnosti je postojanje „SUM" (Sigma) tastera u *toolbar*-u koji olakšava taj posao. Korisnici „Lotusovih" *spreadsheet*-ova neće imati problema jer je na raspolaganju i alternativni meni u „Lotus" stilu sa poznatim rasporedom komandi.

Kontrola izgleda tabele je kompletna. Tu je uključena mogućnost poravnavanja teksta sa marginama, zadebljana (*bold*) slova, kurziv (*italics*), različiti tipovi fontova, boja i veličina. *Excel* koristi fontove iz *Windows*-a, pa je tako moguće koristiti i YU fontove što sa ostalim *spreadsheet*-ovima nije slučaj. Interesantna je mogućnost automatskog podešavanja širine kolone prema najširem podatku u ćeliji. Naslove ili neki tekst možete staviti u posebne boksove koji se lako premeštaju ili se njihova veličina lako menja.

Makroi se mogu raditi na dva načina: pisanjem ili preko makro rekordera. Relativno je jednostavno raditi sa njima, ali tu ima malo sličnosti sa makroima iz *1 2-3*. Kao i *Quattro Pro*, i ovde možete makro proceduru pridružiti 3D „tasteru" a njegova pozicija može biti bilo gde na ekranu. Mislim da je to jako korisno posebno ako pravite aplikaciju za korisnike koji ne znaju *Excel*.

I *Excel* poput „Lotusa" ima *Solver*. Rekao sam da on služi za analizu podataka odnosno za pronalaženje najboljih rezultata do kojih se dolazi promenama varijabli. Grafika je što se može i očekivati od pravog *Windows* programa na zavidnom nivou. Grafikoni se lako kreiraju. Jednostavno označite u tabeli oblast sa podacima i aktivirate odgovarajuću ikonu/taster sa *toolbar*-a. Grafik možete vući i pozicionirati bilo gde u tabeli. Ako kliknete dva puta na grafikon on će se pojaviti u punoj veličini a dobićete i **graph** meni pomoću koga možete doraditi vaš grafikon. Ceo taj posao je vrlo zanimljiv a rezultati su veoma dobri.

„Microsoft" je sa *Excel*-om napravio dobar posao. Program je odličan a koliko znam dobro se prodaje. Za sada nema konkurencije mada se konkurenti trude da mu dođu glave. Tu pre svega mislim na *Informix Wingz* i *Lotus 1 2 3/W*. Ima *Excel* i nekih malih nedostataka koji se pre svega ogledaju u tome da je za njegovu upotrebu potreban dosta jak hardver. Možda je nekome značajno i to da se *Excel* nije dobro pokazao kada radi u mreži za razliku od DOS *spreadsheet*-ova. Na kraju još jedna sitnica: uputstva za upotrebu nisu na nivou onih koje imaju „Lotus" i „Borland".

NA BRZINSKOM TESTU

Da bih uporedio način rada, brzinu i neke kvalitete ovih programa koristio sam svoje test tabele koje inače koristim u ovakvim i sličnim situacijama. Već sam rekao da mislim da je *1 2 3* standard za ovu vrstu programa pa su i test tabele urađene u *1-2-3* verzije *2.3*. Obzirom da su se obe verzije *1-2 3* o kojima ovde govorimo pojavile posle *Quattro Pro* i *Excel*-a, normalno je da ima funkcija i makro komandi koje oni ne podržavaju. U tom smislu sam preradio test tabele. Namera mi je bila da utvrdim kompatibilnost svih *spreadsheet*-ova sa WKJ formatom jer svi (bukvalno) proizvođači *spreadsheet*-ova vode računa o tome i u reklamama navode da su njihovi programi kompatibilni sa *1 2 3*. Drugo, hteo sam da vidim kako i kojom brzinom se vrše neke osnovne radnje u tabeli





Suveren u Windows svetu *Microsoft Excell 3.0*

karakteristične za *spreadsheet* programe uopšte, i treće, obzirom na sve značajniju i traženiju mogućnost kvalitetne grafičke prezentacije podataka interesantno je da se vidi kako tu stvari stoje.

Što se tiče kompatibilnosti situacija je sledeća: sva četiri programa su 100% kompatibilna sa *1 2-3* verzija *2.01*, a *Quattro Pro* i *Excel* su delimično kompatibilni sa *1-2-3* verzijama *2.3* i *3.1+*. Već sam rekao zašto je to tako a najavljene nove verzije ovih programa će sigurno otkloniti ove probleme. Za ispitivanje sam koristio dve test tabele. U jednoj je mala aplikacija sa menijem iz koga se može obaviti par standardnih poslova vezanih za *spreadsheet* kao što su unos podataka, pregled, pretraživanje, štampanje izveštaja i izlaz u DOS. *Quattro Pro* i *1 2-3 v. 3.1+* direktno primaju WK1 format tabela, s tim što *Quattro Pro* meni koji je kreiran makroima, prikazuje u *pull down* stilu. Bitno je da sve funkcioniše bez problema. Što se tiče *Excel*-a, situacija je nešto drukčija. I on prihvata WK1 tabelu ali makro komande morate prevesti kroz *Macro Translator* program i tek onda ih koristiti. Najinteresantnije je da onda imate dve tabele: jednu sa podacima a drugu sa prevedenim makro procedurama. No, sve radi odlično. *Excel* prihvata i WK3 format (iz *1 2-3 v. 3.x*) samo što umesto trodimenzionalnih tabela pravi više nezavisnih ali linkovanih tabela. Kad smo već kod toga, *Excel* dozvoljava linkovanje svojih sa *1 2-3* tabelama. Druga test tabela sadrži sve raspoložive funkcije iz *1-2-3* verzija *2.3*. Kao što rekoh, i *Quattro Pro* i *Excel* su bez problema prihvatili sve @ funkcije koje je sadržao *1-2-3* verzija *2.01*. Za ostale su prijavili @ ERR ili N/A. Iz svega ovoga zaključujem da su i *Quattro Pro* i *Excel* kompatibilni sa onom verzijom *1 2-3* koja je bila aktuelna u vreme njihovog pojavljivanja na tržištu. Dakle, korisnici *1 2 3* koji žele da pređu na rad sa nekim od ova dva programa ne bi trebalo da imaju većih problema pri upotrebi već postojećih tabela.

U drugom delu testa poredio sam brzinu izvršavanja nekih karakterističnih komandi koje postoje u svim *spreadsheet* programima a vrlo se često koriste. To su komande **Copy**, **Move**, **Save**, **Retreive** (**Open**) i **Sort**. Isto tako značajan je i podatak o brzini preračunavanja tabele. Rezultati se mogu videti na grafikonu. Treba napomenuti da je veličina tabele oko 100 K a da su merenja rađena u tabelama originalnih formata. Znači izvorna WK1 tabela je ubačena u svaki od programa a onda snimljena u sopstvenom formatu. Tek nakon toga sam vršio test. Kao što vidite, rezultati su zanimljivi. Na prvi pogled izgledaju dosta ujednačeni i stvarno, koma je bitno da li se neka komanda izvršava za jednu ili tri sekunde? No, to je samo na prvi pogled. U toku višesatnog i višednevnog rada te stvari i te kako dolaze do izražaja. U opisu programa sam rekao da sporost *Quattro Pro*-ovog interfejsa ubija. To je utisak stečen u radu mada kad pogledate rezulta



Uravnoteženo na papiru, drastično u praksi. Rezultati uporednih testova programa *Lotus 1-2-3 2.3, Lotus 1-2-3 3 1 Quattro Pro 3.0* i *Excel 3 0*

Tabela 1 Uporedni prikaz osobina i pogodnosti

Mogucnost	123 2 3	123 3.1+	QP 3 0	ME 3 0
Minimalni RAM (K)	384	1024	512	>1024
Extended memorija	Da	Ne	Da	Da
Expanded memorija	Ne	Da	Ne	Ne
Hard disk	Da	Da	Da	Da
Kolone/Redovi	256/8192	256/8192	256/8192	256/16384
Broj povezanih datoteka	Ogr mem	Ogr mem	Ogr mem.	Ogr mem
Maksimalni broj prozora	1	3	+	+
Više tabela u datoteci	Ne	256	Ne	Ne
Ubacivanje grafikona u tabelu	Da	Da	Da	Da
Štampanje u pozadini	Da[4]	Da	Ne	Da
Makro biblioteka	Da	Da	Da	Da
Makro dibager	Da[5]	Da[5]	Da	Da
Pouzdanost programa	10	10	8	1[6]
UAE greska[6]	Ne	Ne	Ne	Da
Bagovi u programu	Ne	Ne	Da	Ne
Podrske nez. proizv. softvera	Da	Da	Ne[7]	Ne
Broj Add-In programa	>500	>500	Ne[7]	Ne
Razvojni modul (Add-In Toolkit)	Da	Da	Ne	Ne
Kompajler	Da	Da	Ne[7]	Ne
WYSIWYG	Da	Da	Da	Da
Prikaz fontova na ekranu	Da	Da	Da	Da
Linije i senčenje ćelija	Da	Da	Da	Da
Pridruživanje atributa pomoću	Karakter	Karakter Oblast	Oblast Globalno	Karakter Oblast
Globalno				
Rad sa mišem	Da	Da	Da	Da
Interaktivni dijalog boksovi	Da	Delimično	Da	Da
Izbor boja iz palete	224	16	16	71
Uvoz i pregled PIC i CGM dat	Da	Da	Ne	Da
Rotiranje grafičkih objekata	Da	Ne	Ne	Da
3D grafički efekat	Da	Da	Da	Da
Pravi 3D grafikoni	Ne	Ne	Da	Da
Mreža na tabeli	Da	Da	Da	Da
Podešavanje boja	Da	Da	Da	Delimično
Poravnavanje teksta	Da	Da	Ne	Da
Centriranje teksta u kolonama	Da	Da	Ne	Da
Kopiranje formata ćelija	Da	Da	Ne	Da
Biblioteka fontova	Da	Da	Ne	Da/Windows
Sopstveni format stilovi	Da	Da	Ne	Da/Windows
Broj fontova po tabeli	8	8	8	>8
Tekst stilovi po tabeli	>128	8	8	>128
Upotreba boja	Tekst Tekst	Tekst Pozadina	Tekst Pozadine	Pozadina
UNDO	Da	Da	Da	Da
Lak za upotrebu[1]	10	6	1	6/8[2]
Lak za programiranje[1]	10	8	7	1
Sortiranje (brzina/br ključeva)[1]	6	6	8	8
Korisnički interfejs (izgled)[1]	6	6	8	10
Vreme potrebno za obuku	10	8	8	1[3]
Dokumentacija[1]	10	10	8	5

[1] Subjektivna procena, najviša ocena 10, najniža 1
[2] Ako je korisnik već radio sa *Windows*-ima onda je ocena 8
[3] Računajući i *Windows*
[4] Aktiviranjem posebnog programa BPRINT pre učitavanja samog 1-2-3
[5] Samo Single Step
[6] Česta greška kod *Windows* programa (*Unrecoverable Application Error*)
[7] Osim toga je tabela .WK1 kompatibilna

te merenja razlike i nisu tako velike. Činjenica je da kle da je u većim operacija vezanih za aranžiranje podataka u tabeli *Quattro Pro* najsporiji. Koliko utisak u toku rada može da prevari govori i iskustvo sa *1-2-3* ver *3 1+*. Uvek sam mislio a i posle testa mi je teško da se razuverim da je mnogo sporiji od *1 2-3* ver *2 3* pa čak i od ver *2 01*. No rezultati govore da je brzine izvršavenja komandi među ovim verzijama veoma izjednačena. Prilično me je iznenadio rezultat *Excel*-a. Obzirom da radi pod *Windows*-ima, znači u potpunom grafičkom okruženju i obzirom na već uvreženo mišljenje o sporosti *Windows*-a, *Excel* radi veoma brzo. To se posebno odnosi na deo komandi za uređivanje tabele.

Vrlo bitna stvar je i brzina preračunavanja tabele. U starim verzijama *spreadsheet*-ova to je bila jedna od najvećih slabosti jer se izračunavanje vršilo dosta sporo. Glavni uzrok tome je bio metod po kome su to programi radili naime preračunavala se uvek cela tabela bez obzira kakve promene u njoj vršili. Sada je situacija dosta drugačija pa većina *spreadsheet*-ova izvršava rekalkulaciju samo onih ćelija koje su u neposrednoj vezi sa izmenjenim-unetim podatkom. Rezultat toga je daleko brži rad. Iz diagrama se vidi da se preračunavanje u *1 2 3* ver *3 1+* i *Quattro Pro* vrši izuzetno brzo dok je kod *Excel*-a rezultat nešto slabiji. No ako se setite prethodne verzije *Excel*-a, onda je ovo izuzetno unapređenje. Ubedljivo najsporiji se pokazao *1-2-3* ver *2 3*. Napominjem da se radi o tabeli koje sadrži isključivo formule i funkcije od oko 100 K. Sigurno je da se rekalkulacija vrši brže u tabelama sa mešanim dakle i brojnim i tekstualnim podacima. Zanimljivo je da na brzinu izračunavanja utiče i način rasporeda podataka u tabeli. Naime za razliku od *Excel*-a kod *1 2-3* i *Quattro Pro* se ono vrši brže ako su podaci raspoređeni po širini tabele to jest, ako zauzimaju više kolona a manje redova.

Posao koji se često obavlja u radu je učitavanje i snimanje tabela na disk. Stoga sam smatrao da je zgodno da vidimo kako tu stoje stvari. Tu su rezultati dosta neujednačeni. Na primer *1 2-3* ver *2 3* najbrže učitava tabelu sa diska, ali je snima dosta sporije od konkurenata. Kod *Excel*-a je obrnuta situacija najbrže snima a sporo učitava. *Lotus 1 2-3* ver *3 1+* je u oba slučaja u zlatnoj sredini što se ne bi moglo reći za *Quattro Pro* koji je ubedljivo najsporiji. Ovde bismo mogli da se osvrnemo i na brzinu startovanja programa. Naime naglasio bih veliku sporost kod *1 2-3* ver *3.1+* pogotovu ako se odmah učitava i WYSIWYG.

Sortiranje podataka u tabeli je takođe česta radnja. Ona se u *1-2-3* može obavljati po dva ključa u *Excel*-u po tri a u *Quattro Pro* po pet ključeva. U testu sam radio sortiranje po dva ključa u svakom programu i kao što vidite rezultati su potpuno ujednačeni se minimalnom prednošću *1-2-3* ver *2 3*.

Na kraju bi iz svega rečenog trebalo izvući neki zaključak. Sve četiri programa spadaju u istu vrstu imaju puno sličnosti naročito u osnovnoj nameni a ipak su dosta različiti. Kroz kratak opis svakog od njih ste verovatno primetili da sam stavio težište na određene mogućnosti koje su mi se činile dobrim. Tako je evidentno da *Quattro Pro* ima zavidne mogućnosti u grafičkoj prezentaciji podataka, čak i u poređenju sa *Excel*-om koji je ne zaboravimo ipak *Windows* program. „*Lotusovi*" *spreadsheet*-ovi imaju veliku prednost u dve stvari najlakši su za upotrebu i imaju ubedljivo najveću podršku od nezavisnih proizvođača takozvanih dodatnih programa (vidi „Računari" 77). Treba dodati i jedinstvenu mogućnost koju ima *1-2-3* ver *3 x* kroz rad u trodimenzionalnoj tabeli. Iz ieste na grupe programa kao što rekoh *Excel* je jedini *Windows* program. Čini mi se da je dobro urađen onako u kompletu. Ima zavidne mogućnosti u analizi podataka programiranju kao i u kvalitetnoj prezentaciji podataka bilo da se radi o grafici ili izveštajima. Uopšte uzev kada se govori o DOS *spreadsheet*-ovima nećete se prevariti ako radite sa bilo kojim od navedenih programa, mada ja lično navijam za *1-2-3* verzija *2 3*. Kod *Windows spreadsheet*-ova situacija je daleko jasnija. I porast konkurencije koja se relativno skoro pojavila (*1-2-3/W*) *Excel* je tu svojim kvalitetima neprikosnoven.

Dakle ako bi trebalo nekoga posavetovati kojim programom od navedenih da radi to ne bi bio lak posao. U svakom slučaju ako radite pod DOS-om i ako ste početnik, preporučujem *1 2-3* ver *2 3*. Ako ste iskusniji korisnik sa većim zahtevima, onda *1 2-3* *3 1+* ili *Quattro Pro*. Ovaj drugi pogotovu ako vam treba kvalitetna grafička prezentacija podataka. Ako ste jedan od onih retkih koji je potpuno prešao na *Windows*-e onda bar za sada dileme nema – *Excel* je rešenje za vas.

MAKSIMIZIRANI DOS

Jedna od osnova „američkog načina života" je teza da je konkurencija dobra za biznis – uvek treba imati (bar) dva takmičara koji će iz dana u dan preticati jedan drugoga i tako zajednički napredovati prema postavljenom cilju. Zato imaju Republikance i Demokrate, Coca Colu i Pepsi Colu, Dinastiju i Dalas, IBM i DEC, Intela i Motorolu... i QEMM i 386MAX! Pošto smo prošlog meseca predstavili QEMM 6.0, ostalo je da se ovoga puta pozabavimo njegovim najljućim konkurentnom u upravljanju memorijom 386MAX.

Dejan Ristanović

Pre skoro dve godine smo u „Računarima 66" detaljno predstavili 386MAX 5.0, neku vrstu (kako smo tada napisali) „međuverzije" koja je tek polovično podizavala upravo objavljeni *Windows 3.0*. U međuvremenu se pojavio 386MAX 5.10 koji je potpuno podržavao novi *Windows* i koji sa zadovoljstvom koristimo već više od godinu dana – iako smo „386 karijeru" počeli QEMM-om, u jednom trenutku smo zaključili da 386MAX nudi za nijansu bolje performanse i, što je mnogo važnije, veću pouzdanost. No, iako nije bio baš na svakom kompjuteru koji smo u međuvremenu „uzeli u ruke" – na nekima je 386MAX 5.10 pravio probleme dok je QEMM radio savršeno, zaključili smo ukratko da uvek treba imati pri ruci i jedan i drugi. No, kako smo na računaru koji svakodnevno koristimo držali 386MAX, veoma nas je interesovala njegova nova verzija.

ZAKRPE U MEMORIJI

Zašto ima toliko problema sa programima za upravljanje memorijom? Uglavnom zato što je sve to jedan veliki „krpež" kada su sredinom 1976. godine *Intelovi* inženjeri završavali razvoj mikroprocesora 8088 i 8086 (koncipirali su ih, svakako, neku godinu ranije), cilj je bio prevazilaženje limita od 64 kilobajta koji su nametali raniji osmobitni mikroprocesori poput 8080, 8085, Z80... I 64 kilobajta su tada tuku na srce bili izuzetno velike cifre tako da je megabajt adresnog prostora koji je 8088 podržavao izgledao sasvim nedostižno. IBM je, dizajnirajući PC, dalje smanjio ovaj prostor tako što je „velikodušno" odvojio 384 kilobajta za BIOS, bejzik interpretator, video memoriju i buduće ekstenzije. Za RAM je ostalo čuvenih 640 kilobajta koji su „postali tesni" mnogo brže nego što se očekivalo, pa su *Lotus* i *Intel* (kasnije se pridružio *Microsoft*) smislili kako da ih prošire – tako je definisan standard za korišćenje tzv. *expanded* memorije nazvan LIM po početnim slovima imena svojih tvoraca. LIM memorija je, na žalost, bila slabo prihvaćena od strane proizvođača AT kompatibilnih *računara* – njihove mašine su imale tzv. *extended* memoriju koja se nije mogla pretvoriti u LIM. Kasnije su se pojavile AT ploče sa specijalnim setom čipova koje mogu da konvertuju *extended* memoriju u *expanded*, ali to i dalje nije bilo „ono pravo" – morali ste se opredeliti za jednu vrstu memorije i svaku promenu „plaćali" rasklapanjem kućišta.

Dizajnirajući mikroprocesor 80386 *Intel* je, pored ostalog, želeo da jednom za svagda reši probleme sa memorijom, obezbedivši njeno slobodno mapiranje u okviru čitavog 4-gigabajtnog adresnog prostora. Pri tom je trebalo obezbediti punu kompatibilnost sa postojećim softverom, što je postignuto uvođenjem tako zvanog virtuelnog moda – onoga momenta kada ste učitali neki program za upravljanje memorijom kao što je 386MAX, vaš PC više nije u pravom smislu vaš – stavljen vam je na raspolaganje tek jedan virtuelni (prividni) PC ali su svi pristupi memoriji pod kontrolom „više sile" koja se u svakom trenutku stara da virtuelni 8086 vidi samo ono što mu je poznato, a de ipak pristupa memoriji od nekoliko megabajta poštujući razne algoritme za upravljanje njom kao što su LIM ili XMS standard. Mora se reći da 80386 (naravno i 80486 kao njegov naslednik) čitav taj posao radi izvanredno i uz gotovo zanemarljiv gubitak performansi – bilo je zapravo neočekivano malo problema sa upravljanjem memorijom dok se nije pojavio *Windows 3.0*.

Glavni krivac za probleme sa *Windows*-om je firma koja ga proizvodi – *Microsoft* – mogao je sebi da dopusti da ne poštuje protokole za alociranje korišćenja i oslobađanje memorije koji su ranije definisali (i prećutno preuzele jedna od druge) tvorci raznih *DOS Extender*-a poput *Quarterdeck*-a, *PharLap*-a i drugih. Čim je *Windows 3.0* nastao, primećeno je da se nikako ne slaže sa tadašnjim verzijama QEMM-a i 386MAX-a. Problem je rešen na prilično neortodoksan način koji smo detaljnije opisali u prošlim „Računarima" – ukratko rečeno, 386MAX i QEMM neprekidno proveravaju programe koji se učitavaju i, kada po nekoj signaturi prepoznaju *Windows 3.0*, naprosto „u letu" preprave ovaj program tako da bude kompatibilan sa njima. Ovakve ideje su prilično slabo usklađene sa potrebom da sistem radi pouzdano, ali šta je tu je: umesto da razmišljamo zašto je i kako nešto „krpljeno" i kakve to probleme može da donese, bolje je da pogledamo šta se dešava u praksi.

GERILSKI NAPAD

386MAX je spakovan izuzetno atraktivno, u povaću kutiju na kojoj je nacrtan neki starogrčki mislilac koji očito lupa glavu raspoređujući memoriju svog PC-ja. Kada jednom „skontate" kako da otvorite kutiju (ima par „pogrešnih puteva" koji će vas, umesto do diskete, „odvesti" do reklama za proizvode firme *Qualitas*), primetićete da je 386MAX prilično „porastao" – nekada se isporučivao na 360 K disketi, onda je prešao na 720 K, a sada je već stigao do 1.2 megabajta, odnosno 1.44 M ako koristite drajv od 3.5 inča. Pošto su datoteke arhivirane, instalirani 386MAX zauzima čitavih 1.7 megabajta na disku, što je bez sumnje mnogo.

Instalacija 386MAX-a nam je ostala u lošoj uspomeni – INSTALL program je, posle zahteva da unesemo ime, prezime i serijski broj (386MAX se nekada predstavljao samo pri učitavanju, a sada i svaka njegova komponenta (386LOAD se puno puta pozove iz AUTOEXEC-a) ispisuje kako svoje ime tako i ime registrovanog korisnika, što znači da se ne morate bojati da ćete zaboraviti kako se zovete), pronašao da već koristimo prethodnu verziju 386MAX-a i, bez i jednog kontrolnog pitanja, napravio pravi rusvaj. Naš stari 386MAX je prepisan u direktorijum OLDMAX.1, zatim je CONFIG „sređen" i najzad smo, pošto je instalacija završena a računar resetovan, došli u situaciju da ništa ne funkcioniše. Trebalo je, naime, umetnuti jednu od opcija vezanih za skenersku karticu; iako smo unapred znali da će nam te opcije biti potrebne, instalacioni program nam jednostavno nije dao priliku da je unesemo! Ostalo je samo da nađemo DOS disketu, resetujemo računar, unesemo potrebne izmene i nastavimo istalaciju – jako napnjajno, pogotovu ako bi posao bio poveren nekom početniku!

INSTALL program nije jedina komponenta naklonjena „spartanskom razgovoru" sa korisnikom: u 386MAX direktorijumu nalazi se tušta i tma raznih .BAT, .COM i .EXE fajlova koje nikako ne treba „na slepo" startovati. Mi smo, na primer, startovali program sa zanimljivim imenom PREINST.BAT i sve je (bez ikakvog kontrolnog pitanja, naravno) vraćeno u stanje pre instalacije 386MAX-a. Tačnije rečeno, AUTOEXEC.BAT i CONFIG.SYS su vraćeni u prethodno stanje, ali stari 386MAX nije „vraćen" iz direktorijuma OLDMAX.1, tako da smo dobili novi 386MAX koji se pozivao na način koji odgovara starom. Ukratko, dobili smo blokiran računar. Srećom, do tada smo već naučili da DOS disketa i nije neophodna: ako dovedete računar u stanje da se ne može butovati, jednostavno pritisnite RESET i tokom inicijalizacije držite pritisnut taster Alt: 386MAX se neće startovati pa bi trebalo da se pojavi DOS prompt. Ako ga nema, DOS disketu treba uvek imati u pripravnosti!

Ukratko rečeno, nekada veoma jednostavna instalaciona procedura koju smo u prošlom prikazu hvalili je iz nejasnih razloga učinjena složenom i prilično rizičnom. Teško da to možemo smatrati korakom u dobrom smeru.

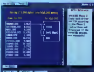
FINO PODEŠAVANJE

Posle osnovne instalacije sledi „fino podešavanje" koje je daleko prijatnije nego kod QEMM-a – umesto da slavljate gomilu kriptičnih skraćenica u jednu jedinu liniju CONFIG.SYS-a, na raspolaganju vam je datoteka 386MAX.PRO u koju pišete sve potrebne deklaracije, komentarišući ih po želji. Konfiguraciona datoteka je uz to omogućila da mnemonici koje 386MAX koristi budu „humanije izabrani", pa je čitav postupak olakšan. Na našem sistemu je u 386MAX.PRO trebalo dodati jedno EXCLUDE=DC00-E000, čime se obezbeđuje normalan rad skenerske kartice; dodali smo i red u kome piše VGA, ne zato što je naročito potreban nego zato što je bio neophodan u jednoj od ranijih verzija, pa smo ostali pri njemu. Ostatak datoteke 386MAX.PRO je kreirao sam instalacioni program.

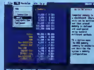
Posle osnovne instalacije sledi, kao i kod QEMM-a, optimizacija tj. „podizanje" rezidentnih programa u UMB (*Upper Memory Block*). Početnike ćemo podsetiti na ranije izrečenu tvrdnju da su IBM-ovi inženjeri, dizajnirajući PC, rezervisali za svoje potrebe prostor između 640-tog kilobajta i prvog megabajta. Rezervisano nije uvek i potrošeno – veći ili manji deo ovog prostora (u zavisnosti od kartica koje koristite) je potpuno slobodan i prosto vapi da ga za nešto upotrebite. No, tu je 386MAX – delovi vaše višemegabajtne *extended* memorije se mapiraju u ove „rupe" i onda se taj prostor koristi na razne načine. Najjednostavnije je objasniti upotrebu samog početka te rezervisane memorije (u *Microsoft*-ovoj terminologiji UMB (*Upper Memory Block*), premda *Qualitas* i dalje koristi termin *high DOS memory*) koji se direktno nastavlja na osnovni RAM – vlasnici *Hercules*-a instalacijom 386MAX-a naprosto dobijaju 64 K osnovnog RAM-a, tako da umesto 640 imaju preko 700 K memorije! Na VGA kartici nema proširenja osim ako ste voljni da se odreknete grafike – primamljiva ideja u raznim prilikama, naročito dok razvijate neki softver koji radi u tekst modu. Autori nove verzije 386MAX-a se, na žalost, nisu dosetili trika koji nudi QEMM 6.0 – uz njega se, naime, isporučuje program VIDRAM koji dopušta da „ukidanje grafike" ne bude trajno već da se aktivira prosim VIDRAM ON i opoziva sa VIDRAM OFF. VIDRAM, na sreću, „sarađuje" i sa 386MAX-om, pa smo tako jednu datoteku iz *Quarterdeck*-ovog kataloga prepisali u konkurentski.

Sledeći trik sa „visokom memorijom" koji 386MAX obezbeđuje je kopiranje ROM-ova u RAM – obzirom da je vreme pristupa EPROM i ROM čipovima u koje je upisan sistemski softver (BIOS, program koji kontroliše disk, video BIOS itd) daleko veće nego vreme pristupa RAM-u, mapiranje delova *extended* memorije u taj prostor i prepisivanje BIOS-a u njega može znatno da unapredi performanse čitavog sistema. Sve, naravno, ima svoju cenu – posle kopiranja ROM-ova u RAM korisniku je na raspolaganju manje *extended* memorije, što je na većini modernih računara sa mnogo RAM-a prihvatljiv kompromis. No 386MAX može da pomogne i onima čiji sistem ima malo memorije – ako vaša ploča rezerviše neki deo memorije za *shadow* RAM ili nešto slično što vam nije potrebno, drajver za upravljanje memorijom će „oduzeti" ovaj RAM BIOS-u ploče i pridružiti ga *extended* memoriji.

Pomenimo i mogućnost podizanja rezidentnih programa u prostor između 640 K i prvog megabajta – svaki slobodan segment UMB-a može da se iskoristi za učitavanje nekog programa koji neprekidno stoji



Program *Maximize* automatski podešava sistem na najveću količinu memorije



Novi grafički korisnički interfejs integriše u jednu celinu sve funkcije programa 386MAX

u memoriji! Takvih programa je zbilja mnogo: bez nekih se jednostavno ne može: drajver za miša, DOSED ili CED, program kojim su definisana YU slova na VGA kartici i odgovarajući drajver za tastaturu, keš program, ponekad čak i *PC Tools* ili *Norton Guide*... Zavisno od hardverske konfiguracije vašeg sistema (IDE diskovi su, na primer, jako povoljni jer praktično nemaju ROM u adresnoj mapi) i vaše veštine, moći ćete da „podignete" dosta rezidentnih programa u UBM i lako uživajući u uslugama koje nude rezidentni programi, i dalje imate slobodnih 630 i više kilobajta osnovne memorije. Jedna od zanimljivih odobina novog 386MAX-a je da i samog sebe podiže u UMB.

Iako smo pomenuli da je za racionalno korišćenje UMB-a potrebno i nešto veštine, firma *Qualitas* se potrudila da automatizuje već deo posla – umesto da „ručno" formirate datoteke AUTOEXEC.BAT i CONFIG.SYS, dovoljno je da pokrenete program MAXIMIZE (ekvivalent programa OPTIMIZE iz paketa QEMM) i izvršavate naloge koje računar daje – nekoliko puta ćete ga resetovati, proveriti ispisane poruke i komentare i... optimalni AUTOEXEC.BAT i CONFIG.SYS su tu! Savršenstvo dopunjeno činjenicom da je iskusnijim korisnicima dopušteno da se „mešaju" u optimizaciju uz korišćenje priloženog editora kojim se „diriguje" podizanjem programa u visoku memoriju: mi smo, na primer, saopštili MAXIMIZE-u da program MARK treba ostaviti tamo gde jeste i taj je zahtev uvažen, što nas je „spasio" ručnog ispravljanja datoteke AUTOEXEC.BAT na koje smo već navikli.

PREKRIVANJE ROM-a

Jedan od dve udarne novitete 386MAX-a 6.0 je dalje proširenje UMB-a putem „prekrivanja" delova ROM-a. Naravno, bitan deo ROM-a se ne sme dirati jer bi pozivanje nekog potprograma iz njega rezultiralo totalnim krahom sistema. Ali, sav ROM zbilja nije bitan – u njemu su i razni SETUP i DIAGS programi koji se eventualno pozivaju pri podizanju sistema a docnije samo nepotrebno zauzimaju prostor koji bi se mogao itekako zgodno upotrebiti. Naravno, glavni problem je u određivanju šta je od ROM-a bitno a šta nije, što nije ni malo lak (a ponekad ni moguć) posao: svaka greška će u nekom nepredvidljivom trenutku izazvati velike probleme smanjujući tako pouzdanost čitavog sistema! Ipak, vredi da se proba.

Čitaoci prošlih „Računara" se svakako sećaju da ovu opciju nudi i QEMM 6.0, ali moramo reći da ju je *Quarterdeck* (po našem mišljenju neopravdano) obavio velom tajne. Tamo se pominju dva takozvana *stealth* mehanizma – jedan bi trebalo da bude „rizičniji" ali da obezbedi više memorije dok je drugi „sigurniji" ali daje nešto slabije rezultate. U okviru dokumentacije nije objašnjeno o kakvim se tehnikama radi niti kakva je razlika između njih – samo se kaže da treba probati prvu, pa ako ne ide preći na drugu. Za razliku od toga, 386MAX (odnosno njegov program MAXIMIZE) naprosto pronađe nekorišćene zone ROM-a i u 386MAX.PRO ugradi niz redova tipa „USE F400-FE00". Dakle, i dalje puna automatika ali se čitav proces nalazi pod punom kontrolom korisnika koji neke od tih zona može, ako ima problema, vratiti ROM-u prostim uklanjanjem odgovarajućeg reda iz konfiguracione datoteke. Na žalost, jevnoća ima svoju cenu – QEMM je na tri sistema koje smo isprobali „izvlačio" 30-50 K prostora u UMB-u **više**. Pa ipak, nama se više svideo 386MAX kod koga smo u svakom trenutku znali šta se dešava... možda i stvar ukusa.

PAMETNA MEMORIJA

I tako stigosmo do glavne namene 386MAX-a – pretvaranja *extended* memorije u *expanded* korišćenjem mogućnosti mikroprocesora 80386/80486. Tu je i drugi udarni novitet 386MAX-a 6.0 (kao uostalom i QEMM-a 6.0 – prosto je neverovatno koliko je napredak ovih programa sličan) – umesto da se *extended* memorija pretvara u LIM, 386MAX je pretvara u nekakvu fiktivnu memoriju koja se zove *SmartMemory*. Ta „pametna memorija" se sama za sebe ne može koristiti, ali može da postane ono što korisniku treba – kada neki program zatraži od 386MAX-a megabajt *extended* memorije, on će mu je dodeliti iz raspoloživog fonda „pametne memorije". Kada neki drugi (ili, što je ređi slučaj, isti taj) program zatraži megabajt *expanded* memorije, 386MAX će odvojiti još megabajt *SmartMemory*, pretvoriti je u LIM i prepustiti programu. Naravno, memorija koju program dealocira vraća se u „nedefinisano" stanje i spremna je za dodeljivanje novim programima koji je zatraže. Zaista elegantno rešenje koje u potpunosti prikazuje prednosti mikroprocesora 80386.

Potreba za ovakvom memorijom zasniva se na činjenici da poslednjih godina sve više programa (*Windows 3.0*, *Excel 3.0*, *AutoCAD* itd) koristi *extended* memorije – LIM standard je, naime, nastao u XT danima koji su davno prošli, a *extended* memorija je na neki način „prirodna" za AT i 386 računare. Pojavio se, doduše, pre nego što su autori softvera mogli (i hteli) da je iskoriste, ali vreme očito radi za nju. Mnogo programa, sa druge strane, i dalje koristi LIM memoriju što znači da u sistemu često treba imati i jednu i drugu – nije malo onih koji, na primer, naizmenično koriste *Windows*, *Lotus*, *WordPerfect*, *Venturu*... Proširenje RAM-a na 8 ili 16 megabajta (čime bi se obezbedilo dovoljno svake vrste memorije) nije baš jeftino, stalno menjanje CONFIG.SYS-a i datoteke 386MAX.PRO je zamorno (čak i ako ga automatizujemo nekim programom kao što je BOOT.SYS) dok 386MAX 6.0 predstavlja zaista idealni kompromis – programu koji traži memoriju se uvek dodeljuje baš ono što mu je potrebno!

Najneprijatnije iznenađenje 386MAX-a 6.0 je saznanje da nije kompatibilan sa *Windows 3.0* u standard režimu. Zarad početnika treba da kažemo da *Windows 3.0* može da radi u *real*, *standard* ili *386* modu – prvi režim je grubo rečeno namenjen XT računarima, drugi mikroprocesoru 80286 a treći, kako samo ime kaže... *Real* režim se ponekad koristi za pokretanje starih *Windows* aplikacija dok u normalnim uslovima *Windows* radi u 386 modu. Nekompatibilnost sa *standard* režimom smo, istini za volju, otkrili čitajući dokumentaciju pošto u praksi nikada ne koristimo WIN /S (a i zašto bismo?). Pa ipak, neki korisnici su utvrdili da *Windows* aplikacije rade nešto brže u *standard* režimu pa ga povremeno koriste – za njih će ova nekompatibilnost sa 386MAX-om biti priličan hendikep na koji korisnici QEMM-a nisu „osuđeni". Stvar je još čudnija ako se zna da je prethodna verzija 5.1 savršeno sarađivala sa svim režimima *Windows*-a, kako to da se *ded* te kompatibilnosti naprasno „izgubi"? U dokumentaciji se, doduše, pominje da će *386MAX možda biti kompatibilan sa budućom verzijom Windows-a u standard režimu* (vrlo čudna konstrukcija, zar ne?) što bi možda moglo da se protumači kao pravovremeno prilagođenje *Windows*-u 3.1. Na žalost, obzirom da *Windows 3.1* još nije izašao a eksperimentisanje sa raznim beta verzijama retko daje rezultate u koje se čovek može pouzdati, ostaje da nestaganje sa *Windows*-om tretiramo kao ozbiljan nedostatak 386MAX-a 6.0!

Pouzdanost 386MAX-a smo ispitali na šest različitih računara i prilično smo zadovoljni rezultatima. Obzirom na kratko vreme koje nam je bilo na raspolaganju, test smo sveli na standardni MAXIMIZE i probanje programa *Windows*, *WordPerfect*, *Lotus 3.0* i *Ventura Publisher*. Samo smo na dva računara imali problema, pri čemu je jedan problem rešen dodavanjem EXCLUDE direktive za neku zonu memorije koju je koristila specijalizovana AD/DA kartica. Na drugom računaru je sam 386MAX radio sasvim korektno, ali je startovanje nekih programa iz paketa (više o njima docnije) proizvodilo poruku XMS/EMS Sharing Error i blokiranje sistema; ova poruka nije ni pomenuta u okviru dokumentacije pa sa njom nismo uspeli da se izborimo. Pa ipak, ovi rezultati su znatno povoljniji od onoga što se (na istim računarima) dešavalo dok smo testirali QEMM. Dodajmo na to činjenicu da 386MAX prema našim merenjima postiže za nijansu bolje performanse (tj. za par procenata manje usporava sistem nego QEMM) i... eto prilike da preporučimo proizvod firme *Qualitas*. Da sve ne bude jednostavno, postarala se već pomenuta činjenica da QEMM u većini slučajeva, uz primenu *stealth* opcije, obezbedi više memorije u UMB-u!

GOMILA ALATKI

Paket 386MAX ne čini samo glavni program za upravljanje memorijom – u onih 1.7 megabajta ulazi i niz uslužnih programa i alatki za analizu sistema. Većina ih je integrisana u *shell* koji se zove MAX mada je u svakodnevnom radu često pogodnije koristiti komandu 386UTIL čijim se opcijama dobijaju razne usluge: 386UTIL /L, na primer, prikazuje kompletnu raspodelu memorije, sa 386UTIL /S se prikazuju podaci o rezidentnim programima, 386UTIL /T testira brzinu memorije i tako dalje. Tu je zatim zanimljiv program SNAPSHOT koji prepisuje kompletnu organizaciju sistema u tekst fajl koji, na primer, možete poslati na Sezam ako se ne snalazite sa raznim tipovima memorije i želite da vam neko iskusniji pomogne: u datoteci su razne informacije koje će svakako olakšati lociranje problema.

Qualitas je odlučio da konačno parira i *Quarterdeck*-ovom programu MANIFEST koji se isporučuje uz QEMM 6.0. Radi se, zapravo, o programu koji analizira vaš sistem uočavajući razne parametre vezane za raspodelu i funkcionisanje memorije i generiše razne izveštaje i sugestije koje vam omogućavaju da podesite što se može podesiti da biste iz svog hardvera izvukli maksimum. Naravno, za razumevanje **svih** podataka koje *Manifest* kreira treba dosta toga znati a za „fino podešavanje" treba malo i dosta iskustva, ali će čak i početnik uz pomoć ovog programa saznati osnovne stvari o svojoj PC konfiguraciji.

Približni ekvivalent MANIFEST-a je program ASQ koji se isporučuje uz 386MAX. Startujte ga, sačekajte da se završi analiza sistema a onda kroz nekoliko menija pregledajte stavke koje vas zanimaju. Moramo ipak reći da nam se MANIFEST učinio daleko pogodnijim i logičnijim, možda zato što smo se na njega navikli. Neprijatna osobina ASQ-a je što na pomenutom računaru koji je prijavljivao tajanstvenu poruku *XMS/EMS Sharing Error* nismo uspeli ni da ga startujemo – stalno je davao poruku MS Error pri čemu je ono X verovatno „progutala" leva ivica ekrana. Nije baš zgodno kada se alatka koja treba da omogući izolovanje problema *ne može* ni pokrenuti zbog samog problema koji treba da reši!

U paketu dobijate i poseban keš program kao i drajver za RAM disk, navodno su posebno prilagođeni programu 386MAX. Radi se ipak o sasvim standardnim proizvodima u svojoj klasi – ništa što bi nas naterало da menjamo naš omiljeni *PC Kwik*.

386MAX 6.0 najzad obezbeđuje i vlasnicima AT računara zasnovanih na NEAT i nekim drugim čipovima podizanje rezidentnih programa u UMB: u osnovnu cenu paketa je, dakle, uključen približan ekvivalent *Quarterdeck*-ovog QRAM-a koji se ranije morao posebno kupovati. Ovu komponentu paketa nismo testirali – možda će biti prilike u nekom od sledećih brojeva „Računara".

Ukratko, 386MAX 6.0 je ostavio povoljan utisak: veoma smo zadovoljni njegovom pouzdanošću i performansama, zameramo mu razna „sitnice" oko instalacije i uslužnih programa, kao i nekompatibilnost sa *Windows 3.0 standard* režimom. Ukoliko imate 386/486 računar koji tek treba opremiti programom za upravljanje memorijom, vredi početi eksperimentisanje od 386MAX-a 6.0!

PROGRAMIRANJE OD OKA

Bejzik pod *Windows*-ima, na šta bi to moglo da liči? Kompatibilan sa drugim PC bejzicima, ali svakako i prilagođen novoj sredini. Kako pomešati bejzik sa prozorima? Da li očuvati kompatibilnost po svaku cenu, a od *Windows*-a šta ostane!? Ili suprotno, obogatiti jezik kompletnom podrškom za *Windows*-e, a od bejzika šta ostane!? „Korak napred, dva koraka nazad" ili „dva koraka napred, jedan unazad"!? Majkrosoft se odlučio za ovo drugo: vizuelni bejzik je pre svega sredstvo za savladivanje čudne radne sredine po imenu *Windows*, s tim što se klasične računske operacije izvode „kao u bejziku".

Duška Savić



Windows-i su do sada prodati u neverovatnih devet miliona primeraka, a intenzitet prodaje ne jenjava – naprotiv. Umorni DOS zamenjen je veselim sličicama, mišem i bojama, a i korisnim programima svih vrsta, od procesora reči do emulacije VAX-a. Neke softverske kuće, a među njima Lotus i WordPerfect, u prvi mah potcenile su značaj ove nove platforme. Vremenom su se priključile trendu posebnim verzijama svojih glavnih paketa. Talas oduševljenja potpuno novim tržištem osetio se i među proizvođačima sistemskog softvera, pa skoro svi (osim firme JPI) nude nova verzije jezika za *Windows*-e. Uglavnom postoje dve vrste ponude: posebne verzije istog jezika za DOS i *Windows* (*Turbo Pascal 6.0* i *Turbo Pascal for Windows*) ili oba jezika u istom paketu (*Borland C++*, *MS C/C++ 7.0*). *Visual BASIC* pripada prvoj grupi jer je njegov stariji brat, GWBASIC, od samog početka bio isporučivan kao sastavni deo DOS-a.

KROZ PROZOR... NA TRŽIŠTE

Mnogi su se čudili što su *Windows*-i uopšte pušteni u prodaju bez sopstvenog bejzika. Odgovor treba tražiti u strategiji Majkrosofta. Oni već nekoliko godina prodaju poseban paket po imenu *SDK for Windows*. U njemu je sve što je potrebno da se prave programi pod *Windows*-ima – pod uslovom da znate ili volite C. Prodato je 65.000 primeraka tog paketa, ali niko nikad nije rekao da je s njim lako raditi ili pisati programe. To je konačno shvatio i sam Majkrosoft, tako da u novoj verziji njihovog C-a (koji je istovremeno i C++) značajan deo ima SDK. (Sam *SDK for Windows* će se i dalje prodavati odvojeno jer neke firme i dalje uz svoje prevodioce ne isporučuju dovoljno komponenti da bi se sa njima pisali programi za *Windows*-e, primer za to je C++ firme JPI.) I dok originalni SDK gubi na značaju, odnosno postaje „opšte dobro", drugi jezici su odavno počeli da nude mogućnosti pisanja programa direktno pod *Windows*-ima. Jedan od prvih je bio *Smalltalk/V for Windows*, a poznati objektni jezik *Actor* da i ne spominjemo. Nedavno je i Borland verzijom 3.0 svog C i C++ paketa definitivno „prošao kroz prozor". Tu je Majkrosoft bio suočen sa očiglednim nedostatkom svoje ponude: novi operativni sistem nije imao svoj bejzik! To su uočili i drugi, pa su se pojavila još dva bejzika za *Windows*-e. Majkrosoftov odgovor na to je bio *Visual BASIC*.

Strateški gledano, *Visual BASIC* je ključna karika Majkrosoftove ponude: glavne vrste primena za *Windows*-e (obrada teksta, unakrsne tabele itd.) zastupljene su vrlo jakim aplikacijama, ali uspeh u poslovnom svetu zavisi i od programa za vertikalna tržišta, tj. od programa koje pripremaju programeri po preduzećima. A ti „programeri" nikako ne moraju biti profesionalci naoružani C-om, objektima, CASE alatima i drugom teškom artiljerijom. BASIC je kao stvoren za ne-profesionalne programere. *Windows*-i su kao stvoreni za ne-profesionalne korisnike. Ukratko, bejzik je stub temeljac svakog operativnog sistema jer pokazuje šta se sve može napraviti u datoj radnoj sredini, obuka ne traje dugo, strogo interaktivna priroda deluje umirujuće na početnike... previše kvaliteta da bi bio ignorisan. Tako dolazimo do očigledne marketinške ideje: navesti i ljude koji ne znaju neka naročite programska jezika da se preko onoga što svi znaju – a to bi bio bejzik – upute u programiranje pod *Windows*-ima. Bejzik za *Windows*-e treba da garantuje i poslednjem nevernom Tomi da će se kad-tad pojaviti baš ona aplikacija koja mu život znači (a inače je toliko beznačajna da za nju ne postoji komercijalni paket).

INSTALACIJA I PAKET

Visual BASIC se isporučuje na dve diskete prečnika 5.25 inča i kapaciteta 1.2 megabajta. Dokumentacija se sastoji iz dve knjige: *Language Reference* (350 strana) i *Programmer's Guide* (456 strana). Prva knjiga detaljno opisuje sve naredbe, a druga je uvod u osnovne elemente i ideje programiranja sa vizuelnim bejzikom. Ime programa dosta dobro opisuje ideju i nameru izdavača: napraviti bejzik sa kojim će korisnik raditi više vizuelno nego proceduralno.

Izdavač je Microsoft Corporation, One Microsoft Way, Redmond, WA 98052-6399, USA. Cena je 199 USD, a u maloprodaji se može naći za 165 USD.

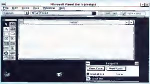
Instalacija je jednostavna i svodi se na aktiviranje programa SETUP i umetanje disketa po redu. Po izvršenoj instalaciji, vizuelni bejzik se prijavljuje kao posebna ikona u prozoru *Windows Applications*. Klik nad tom ikonom vodi u sam program, tj. aktivira bejzik. Uobičajeno je da se Windows aplikacija izvršava u sopstvenom prozoru, a da u okviru njega organizuje potprozore ako za to ima potrebe. Vizuelni bejzik tu odstupa od pravila. Na vrhu ekrana nalazi se nešto što bi bilo zaglavlje normalnog prozora – ime, meni, još jedan red sa ikonama, ali okvir i površina tog novog prozora ne pokrivaju ekran: istovremeno sa prozorima vizuelnog bejzika na ekranu ostaju i vide se „u pozadini" svi prozori koji su se tu zatekli. Ako baš želimo da gledamo samo u prozore vizuelnog bejzika, moramo *Program Manager* pretvoriti u ikonu.

GLAVNI MENI

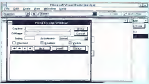
Glavni meni se sastoji iz šest opcija: **File**, **Edit**, **Code**, **Run**, **Window** i **Help**. Opcija **File** bavi se upisom i čitanjem projekata, formulara, modula i samih datoteka. Opcija **Edit** je skoro istovetna sa istoimenom opcijom u svim drugim *Windows* programima: **Cut**, **Paste**, **Copy** itd. U opciji **Code** nalaze se naredbe za razgledanje i čitanje teksta programa, kao i indikator da li da se tekst kojeg programer unosi sintaksno analizira u smislu bejzik jezika. Ako da, onda se stiče utisak totalne interaktivnosti: sistem upozorava na sintaksne greške tokom unosa programskog teksta, baš kao nekad na osmobitnim računarima. Ostale subopcije za **Code** odnose se na pretraživanje teksta programa i brzo prelaženje od procedure do procedure.

Opcija **Run** deli se na **Start** (početak programa), **End** (kraj programa), **Restart** (ponovno izvršenje), **Single step** (izvršenje programa korak po korak), **Procedure step** (izvršenje programa proceduru po proceduru), **Set Next Statement** (izvršenje se nastavlja od tekuće linije), **Show Next Statement** (pokazuje koja će se sledeća naredba izvršiti), **Toggle Breakpoint** (postavlja ili uklanja zaustavnu tačku (*breakpoint*) u programu), **Clear All Breakpoints** (briše sve zaustavne tačke), **Set Startup Form** (zadaje kojim formularom će početi aplikacija), **Modify Command$** (omogućava analizu programa koji koristi naredbu **Command$**).

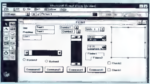
U *Windows* programima obavezna je opcija **Window**, ali nije naročito značajna: obično samo raspoređuje potprozore na različite načine. U vizuelnom bejziku ona je bogatija jer tu su naredbe za izbor boja (**Color Palette**), otvaranje interaktivnog prozora (**Immediate Window**), otvaranje prozora za oblikovanje menija (**Menu Design Window**), otvaranje prozora za rad sa projektima (**Project Window**) i otvara-



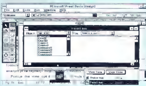
Ekran vizuelnog bejzika sa prozorima *Project* (dole desno), *Toolbox* (krajnje levo) i nepopunjenim formularom *Form1* (u sredini). Ispod glavnog menija je meni sa vertikalnim spiskom osobina formulara (*Caption*) i, desno od njega, sadržajem osobine *Caption* (*Form1*)



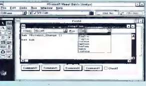
U prozoru *Project* aktivirana je opcija *View Code*. Pojavljuje se poseban prozor u koji upisujemo „obične" potprograme. Na levoj strani vidi se spisak upravljačkih objekata koji su u formularu *Form1*, a sa desne strane bi se video spisak procedura.



Nasumice postavljeni objekti iz kojih se sastoji korisnički interfejs u vizuelnom bejziku



Potprozor za generisanje menija



Na objektu tipa *VScroll* kliknuto je dva puta mišem. Kao rezultat, pojavljuje se poseban prozor u kojem se pišu procedure koje reaguju na događaje. Na desnoj strani vidi se spisak mogućih događaja, a u prozoru se nalazi već pripremljeni kostur procedura, sa imenom *VScroll1_Change*.



Jednostavna aplikacija za konverziju jednog rasporeda YU slova u drugi. Donji levi prozor predstavlja prozor sa tekstom klik procedure, a dole desno se vidi prozor tipa PROJECT.

nje palete sa alatima za pravljanje interfejsa aplikacije (**Toolbox**)

Opcija, **Color palette** omogućava izmenu boja formulara ili upravljačkog objekta. Moguće je stvarati nove vrste boja, kao kombinaciju tri osnovne boje (*dithering*)

REŽIMI RADA

Postoje tri režima rada

design time, tj. programiranje – pravljenje aplikacije. Ono se deli na pravljenje formulara (raznih programskih maski i prozora) i na pisanje naredbi kojima se odgovara na događaje.

run time izvršenje programa, onako kako bi ga korisnik video,

break mode, prekid izvršenja programa. Tu je moguće ispitivati razna programska stanja, menjati tekst programa, izvršavati ga ponovo od početka ili od neke posebne tačke u programu i slično

CRTANJE PROGRAMA

U klasičnim jezicima programi se pišu, a u vizuelnom bejziku programi se *prvo* crtaju, pa pišu. Vizualni bejzik nije generator koda u smislu da korisnik na ekranu nacrta ulazne i izlazne maske, pa da se posle na disku stvori datoteka sa naredbama na bejziku, koja u stvari predstavlja novi program. Ovde korisnik crta maske postavljanjem upravljačkih objekata u okviru jednog ili više prozora (tzv. formulara). Izvršni deo bejzika u sebi pamti sve interakcije i interpretira korisnikove izbore kao pozive odgovarajućih potprograma.

Za razliku od drugih generatora koda, ovde se na ekran postavljaju „živi" objekti koji mogu da odgovaraju na razne stimuluse i na razne načine. Neophodno je vizuelnom bejziku saopštiti koji su to stimulusi i koje su to reakcije. Ovo drugo se izvodi pisanjem naredbi na bejzi-

ku (konačno stigosmo i do bejzika!) a ono prvo se radi interaktivno, takođe u fazi oblikovanja formulara.

Ovaj recept bi se mogao primeniti i na sve druge jezike sa kojima se pišu interaktivni programi.

OBLIKOVANJE FORMULARA

Prozor tipa **Toolbox** je skup od 16 ikona koje se aktiviraju klikom miša i – osim jedne – služe za unošenje upravljačkih objekata u formular. Ta jedna, u obliku strelice, nalazi se u gornjem levom uglu prozora i služi za promenu veličine i položaja već unetog upravljačkog objekta. Ostale ikone i istovremeno vrste upravljačkih objekata su:

- Slika (**picture box**), može biti pasivna ili aktivna, složena iz niza tačaka (*bitmap*), ikona ili tipa *metafile* (zapisuju se kompletni objekti a ne samo nizovi tačaka). Slika može biti uvezena iz datoteka na disku (npr. tipa BMP ili ICO), nacrtana naredbom **Print**, ili uvezena iz neke druge *Windows* aplikacije putem DDE *razmene*.
- Konstantni tekst (**label**), tekst kojeg u normalnom toku programa korisnik neće menjati (ali ga aplikacija može promeniti). I konstantni tekst se može promeniti kroz DDE.
- Linijski editor (**text box**), sadrži tekst kojeg korisnik može (a po pravilu i treba) da promeni. Naravno, i aplikacija može menjati sadržaj u linijskom editoru. U bejzik terminima, sadržaj tog polja je običan string, pa za numeričke podatke treba primenjivati naredbu **Val**. I sadržaj editora se može promeniti kroz DDE.
- Okvir za upravljačke objekte (**frame**), grupi se upravljačke objekte u manje celine u okviru formulara. Posle toga je moguće, na primer, pomeranjem okvira pomerati i sve objekte koji su u njemu.
- Ekranski taster (**button**), aktivira se klikom miša ili pritiskom na odgovarajuće tastere na tastaturi. Grafički, posle aktiviranje ekranski taster se menja i zadobija izgled kao da je stvarno bio pritisnut (odatle mu i ime). Kroz ovu vrstu upravljačkih objekata korisnik programe direktno zadaje naredbe: bez ovakvog upravljačkog objekta, krajnji program ne bi bio interaktivan.
- Višestruki izbor (**check box**), postavlja pred korisnika najmanje dva izbora, npr. da li da se neki indikator uključi ili ne, da li da se štampa ili ne i slično. Može se upotrebiti kad god korisnik treba da postavi jednu ili više vrednosti istovremeno.
- Jednostruki izbor (**option button**), korisnik bira jednu jedinu opciju iz niza ponuđenih mogućnosti. Izbor jedne poništava sve prethodne, što je slično biranju stanica na radio aparatu. Zato se ova vrsta upravljačkih objekata sreće se i pod nazivom „radio taster".
- Kombinacija teksta i ponuđenog spiska vrednosti (**combo box**), korisnik ili unosi vrednost kao u linijskom editoru ili bira iz spiska ponuđenih vrednosti.
- Ponuđeni spisak vrednosti (**list box**), korisnik bira mišem jednu od ponuđenih vrednosti.
- Horizontalna skala sa proporcionalno raspoređenim markerom položaja kursora u odnosu na celinu teksta (**horizontal scroll bar**), položajem markera može se manipulisati odvojeno od teksta, što za posledicu ima i premeštanje kursora po tekstu.
- Vertikalna skala sa proporcionalnim markerom (**vertical scroll bar**), slično, samo što se kursor pomera ka početku ili ka kraju teksta.
- Zadavanje događaja u vremenu (**timer**), upravljački objekat koji zadaje u kojem će se intervalu nešto dogoditi tokom izvršenja programa. Za korisnika nevidljiv.
- Prikaz disk jedinica u sistemu (**drive list box**),
- Prikaz imenika na disku (**directory list box**),
- Prikaz datoteka (**file list box**), poslednje tri opcije služe za stvaranje dijaloga sa korisnikom, na primer da mu omogući da interaktivno promeni disk jedinicu, imenik i ime datoteke.

OSOBINE, DOGAĐAJI I METODI

Pošto se nacrta interfejs za rad sa programom, postavljaju se tzv. osobine (*properties*). Svaki objekat u vizuelnom bejziku ima sopstvene osobine: ime, sadržaj, veličinu, položaj i boju. Abecedno uzev, osobine su ActiveControl, ActiveForm, Alignment, Archive, AutoRedraw, AutoSize, BackColor, BorderStyle, Cancel, Caption, Checked, ControlBox, CtlName, CurrentX, CurrentY, Default, DragIcon, DragMode, DrawMode, DrawStyle, DrawWidth, Drive, Enabled, FileName, FillColor, FillStyle, FontBold, FontCount, FontItalic, FontName, Fonts, FontSize, FontStrikethru, FontTransparent, FontUnderline, ForeColor, FormName, hDC, Height, Hidden, hWnd, Icon, Image, Index, Interval, LargeChange, Left, LinkItem, LinkMode, LinkTimeout, LinkTopic, List, ListCount, ListIndex, Max, MaxButton, Min, MinButton, MousePointer, MultiLine, Normal, Page, Parent, Path, Pattern, Picture, ReadOnly, ScaleHeight, ScaleLeft, ScaleMode, ScaleTop, ScaleWidth, ScrollBars, SelLength, SelStart, SelText, SmallChange, Sorted, Style, System, TabIndex, TabStop, Tag, Text, Top, Value, Visible, Width, WindowState.

Svaki objekat ima niz osobina. Osobine se postavljaju na dva načina: interaktivno pomoću posebnog horizontalnog menija (*properties bar*) ili programski. Meni za osobine sastoji se iz dva polja – za ima osobina (na primer **CtlName**) i za vrednost. Imena osobina vide se kao vertikalni spisak iz kojeg se klikom miša odlučujemo za jednu. U drugo polje upisujemo vrednost. Ako npr. za osobinu **Label** (obeležje) upišemo „Ulazni podatak" na formularu će se odmah videti ta izmena.

Drugi način se zasniva na konvenciji da se osobinama objekta pristupa kao poljima u slogu. Pretpostavimo da smo u formular postavili linijski editor. Njegovih osobina ukupno ima 34. Jedna od njih je **Text** i predstavlja sadržaj linijskog editora. Pretpostavimo da je u polje **CtlName** tog linijskog editora vrednost „UlaznoPolje". To znači da u programu postoji primerak linijskog editora po imenu „UlaznoPolje" i da je to zapravo promenljiva tog tipa. Naredba

UlaznoPolje.Text = „Beograd"

postavlja osobinu **Text** na vrednost „Beograd", pa bi se reč **Beograd** videla u primerku linijskog editora **UlaznoPolje**. Očito da osobine međusobom nisu ravnopravne: **CtlName** sadrži ime u odnosu na koje se sve ostale osobine usmeravaju.

Osobina objekta je promenljiva kao i svaka druga. Na primer, ako je potrebno tekst pretvoriti u broj, izvršili bismo naredbu poput ove:

BrojneVrednost = Val(UlaznoPolje.Text)

Objekat reaguje na događaje. Abecednim redom događaji su Change, Click, DblClick, DragDrop, DragOver, DropDown, GotFocus, KeyDown, KeyPress, KeyUp, LinkClose, LinkError, LinkExecute, LinkOpen, Load, LostFocus, MouseDown, MouseMove, MouseUp, Paint, PathChange, PatternChange, Resize, Timer, Unload. Za svaki objekat postoji tačno određen niz događaja na koje reaguje. Reakcije se sastoji u pozivanju jednog potprograma, koji se u vizuelnom bejziku ne razlikuje previše od potprograma u nekom drugom bejziku. Glavna razlika je u imenu potprograma, koje se ovde određuje automatski i sastoji se iz imena u osobini **CtlName**, crtice i imena događaja. Na primer, potprogram koji se aktivira kada se pritisne neki taster na tastaturi dok je kursor u linijskom editoru **UlaznoPolje** zove se **UlaznoPolje_ClickPress**. Slično, ako želimo da **UlaznoPolje** reaguje na klik mišem ili taster **Tab**, odabrali bi smo događaje **Click** i **GotFocus**, pa bi se u tekstu programa automatski našle i procedure **UlaznoPolje_Click** i **UlaznoPolje_GotFocus**. Naravno, šta će se tačno dogoditi prilikom izvršenja svake od ovih procedura zavisi od toga šta tamo napišemo.

Kao i sve drugo, i ovo kombinovanje imena odigrava se grafički i mišem: ime objekta bira se iz spiska imena upravljačkih objekata koje smo uneli u formular, a ime događaja se bira iz spiska događaja. Unos teksta programa smanjen je na najmanju moguću meru u vizuelnom bejziku.

GENERISANJE MENIJA

Meni aplikacije pravi se iz posebnog prozora. Pomoću njega se postavljaju atributi pojedinačnih opcija, i to:

- **ime opcije**
- **ime upravljačkog objekta** kojim se pokazuje na naredbe koje treba izvršiti kada se opcija odabere
- **indeks** (numerička vrednost koja određuje položaj upravljačkog objekta u nizu takvih objekata),
- **skraćenica** (kombinacije tastera kojom se opcija aktivira),
- **indikator izvršenje menija** (znak koji se pojavljuje pored imena opcije ako je opcija izabrana)
- **indikator za omogućavanje ili onemogućavanje opcije** („omogućena" opcija se može birati, a „onemogućena" ne može i u meniju je izvedena sivkastim slovima) i
- **indikator vidljivosti** (da li da se opcija vidi ili da se ne vidi u meniju).

Osim atributa, u prozoru za pravljenje menija nalazi se i nekoliko „ekranskih tastera" (*buttons*) koji predstavljaju direktne naredbe za opcije. Kursorski tasteri „u levo" i „u desno" menjaju nivo menija (za svaki meni može postojati po pet menija nižeg nivoa), a tasteri „na gore" i „na dole" menjaju položaj opcije u meniju istog nivoa. Vidi se i hijerarhija menija: a meniji nižeg nivoa su uvučeni u odnosu na menije viših nivoa.

Konačno, postoje i ovi ekranski tasteri: **Next** za premeštanje izbora na sledeći red, **Insert** za umetanje reda u hijerarhiju menija, **Delete** za brisanje reda iz hijerarhije, **Done** za zatvaranje celog prozora i prihvatanje svih izmena i **Cancel** za napuštanje prozora ali bez izmena.

VRSTE OBJEKATA

Postoje dve vrste opštih objekata: formulari i upravljački objekti, a postoje i četiri specijalna objekta: **Screen**, **Debug**, **Printer** i **Clipboard**. Za njih postoje specijalne naredbe, takozvani metodi. Abecednim redom, metodi su AddItem, Circle, Clear, Cls, Drag, EndDoc, GetData, GetFormat, GetText, Hide, Line, LinkPoke, LinkRequest, LinkSend, Move, NewPage, Point, Print, PrintForm, PSet, Refresh, RemoveItem, Scale, SetData, SetFocus, SetText, Show, TextHeight, TextWidth. Karakteristika

VISUAL BASIC U AKCIJI

Kao primer najjednostavnije moguće aplikacije neka posluži program koji učitava red po red datoteke ADRESE.DAT, menja slova iz jednog rasporeda YU slova u YUASCII i snima to na disk kao ADRESE.DOC, tako da se može učitati u procesor reči za slanje cirkularnih pisama. Formular koji predstavlja program ima svega tri upravljačka objekta, od kojih su gornja dva, *ime datoteka* i *text1*, za ovako jednostavan posao – suvišni. Da se radi o programu za višekratnu upotrebu, kroz polje *text1* bi se učitavalo ime datoteke koju treba konvertovati. Ovako to ime (adresa.dat) sadržano je u samom programu.

Izvršanje počinje klikom na naredbu „Konverzija teksta". Taj klik automatski poziva proceduru **Command1_Click()**. U njoj se prvo otvaraju datoteke sa logičkim brojevima 1 i 2, za ulaznu i izlaznu datoteku respektivno. U petlji tipa DO UNTIL...LOOP čita se do kraja datoteke (tj. dok funkcija EOF ne vrati vrednost **True**), a u međuvremenu se učitava sa *Line Input* red po red

```
Sub Command1_Click ()
    Open "c:\adrese\adrese.dat" For Input As 1
    Open "c:\adrese\adrese.doc" For Output As 2
    Do Until EOF(1)
    Line Input #1, s$
    nt$ = ""
    For I = 1 To Len(s$)
        a$ = Mid$(s$, I, 1)
        If a$ = "Ž" Then a$ = "Ž"
        If a$ = "Ä" Then a$ = "Š"
        If a$ = "Ö" Then a$ = "Đ"
        If a$ = "Å" Then a$ = "Č"
        If a$ = "Ü" Then a$ = "Ć"
        If a$ = "á" Then a$ = "ž"
        If a$ = "ä" Then a$ = "š"
        If a$ = "ö" Then a$ = "đ"
        If a$ = "å" Then a$ = "ć"
        If a$ = "u" Then a$ = "č"
        nt$ = nt$ + a$
    Next I
    Write #2, nt$
    Loop
    Close #1
    Close #2
End Sub
```

i u FOR petlji zamenjuju znaci. Po izvršenoj konverziji naredbom **Write** se linija piše po datoteci 2. Na kraju se sa CLOSE zatvaraju obe datoteke.

drugim bejzicima. Međutim, ovde postoji i posebna vrsta dinamičkog niza: *niz upravljačkih objekata*. To je niz srodnih objekata koji imaju isto ime i reaguju na iste događaje. Element takvog niza je samostalni objekat sa svojim skupom osobina, s tim da svi elementi imaju istu osobinu **CtlName**. Svaki član niza ima sopstveni redni broj, npr. **KontrolniNiz(1)**, **KontrolniNiz(2)** itd. Vrednost člana ovakvog niza postavlja se posebnim naredbama **Load** i **Unload**.

Najvažnije primene ovih nizova su pisanje menija sa promenljivim brojem opcija i stvaranje novih upravljačkih objekata tokom izvršenja programa.

Niz upravljačkih događaja postoji ali se ne deklariše u programu. Može se stvoriti i tokom izvršenja programa, postavljanjem istog imena u dva različita objekta ili postavljanjem raznih vrednosti u osobinu **index**.

Mogu postojati samo jednodimenzioni nizovi upravljačkih objekata.

Program u vizuelnom bejziku ima strogo modularnu strukturu. Postoji jedan globalni modul i on sadrži deklaracije promenljivih, definicije konstanti i tipova koje treba da razpozna svaki deo aplikacije. Postoji datoteka CONSTANT.TXT u kojoj se nalaze često korišćene konstante, odatle u novi program treba preneti ono što će se koristiti. Te definicije sadrže službenu reč **global**, tako da se konstante moraju uneti u globalni modul (ili prepisati na neki drugi način). Globalni modul ne sme sadržati procedure ili funkcije.

metoda u vizuelnom bejziku je polimorfizam: metod istog imena može se primeniti na bilo koji objekat, i uprkos istom imenu za svaki objekat ima neko drugo dejstvo. Naredba **Printer.Print** štampa na štampaču, naredba **Slika1.Print** štampa na formularu **Slika1** itd.

Objekat **Clipboard** omogućava vezu između vizuelnog bejzika i *Windows*-a: metod **SetText** kopira tekst u **Windows Clipboard** i odatle se taj tekst opcijom **Edit Paste** može preneti u bilo koju drugu *Windows* aplikaciju. Slično, metod **GetText** iz **Windows Clipboarda** preuzima tekst tako da se može koristiti u varijablame vizuelnog bejzika. Objekat **Clipboard** omogućava dinamičku razmenu podataka (DDE) sa drugim *Windows* aplikacijama. Moguće je napraviti program koji će učitavati nove vrednosti dobijene iz nekog komunikacionog programa i sa stalno svežim vestima npr. sa berze crtati aktuelne kurseva valuta ili deonica.

Objekat **Screen** je niz formulara i objekata. Njegova osobine određuju vrstu i broj fontova, širinu i visinu, oblik kursora za miša i – najvažnije – **ActiveControl** i **ActiveForm**, koji otkrivaju upravljački objekat i formular koji su trenutno fokusirani. Izmenom vrednosti u osobini **ActiveControl** moguće je direktno aktivirati svaki upravljački objekat.

PROZOR IMMEDIATE

Prozor tipa **Immediate** se automatski otvara tokom izvršenja programa i tokom prekida rada programa. U njemu se mogu izvršavati pojedinačne naredbe iz programa, npr. mogu biti odštampane vrednosti promenljivih. Naredbe **Debug.Print**, gde je **Debug** specijalni objekat, štampa tekuće programske vrednosti u prozor immediate. Moguće je mišem aktivirati ovaj prozor i direktno iz njega zadavati naredbe **Print**. Stari dobri bejzik – ali u prozoru!

PROŠIRENJA BEJZIKA

Osnovni tipovi podataka su **integer** (dvobajtni celi brojevi, simbol %), **long** (četvorobajtni celi brojevi, &), **single** (četvorobajtni realni brojevi, !), **double** (osmobajtni realni brojevi, #), **currency** (broj sa fiksiranom decimalnom tačkom, @), **string** (niz slova, tekst, $). Ako se ne naznači tip promenljive, podrazumeva se **single**. Tip se može posebno deklarisati i to dodavanjem posebnog znaka iza imena ili rezervisanim rečima **deftype**, **dim**, **global** ili **static**. Najvažnija reč tu je **dim**, jer omogućava deklarisanje imena i nizova. Na primer:

dim I as integer
dim ime as string, pare as currency
dim a(-4 to 10), b(-99 to -5, -4 to 8)

Korisnik može definisati nove tipove podataka naredbom **type** i zatim taj tip koristiti u deklaracijama nizova:

type osoba
ime as string * 9
prezime as string * 15
godina as integer
end type
dim radnik as osoba
dim radnici (1 to 99) as osoba
dim dniz ()

U tipu **osoba** ime je niz od devet znakova, a niz **radnici** ima 99 elemenata tipa **osobe**.

Daleko najznačajnija novina u vizuelnom bejziku su dinamički nizovi. Naredbom **redim** se broj elemenata u nizu može menjati tokom izvršenja programa. Dinamički niz se deklariše pomoću **dim** ili **global**, ali bez navođenja dimenzija. U gornjem primeru **dniz** je dinamički niz. Negde kasnije u programu mora biti izvršena naredba poput

redim dniz(5,6)

Dozvoljeno je redimenzionisati niz neograničen broj puta, ali tu važe i neka ograničenja. Dimenzije se mogu menjati ali ne i jednom ustanovljeni broj dimenzija: trodimenzioni niz ne može postati dvodimenzioni ili obratno.

Nizovi su i dalje ograničeni na ukupnu dužinu od 65535 bajtova, tj. na dužinu segmenta.

Ovako deklarisani nizovi liče na nizove u

PROZOR ZA PROJEKTE

Prozor nazvan **Project Window** prikazuje i organizuje sve delove aplikacije. Zapravo, u njemu se vidi spisak svih formulara i modula. Klikom na imenu iz tog spiska otvara se odgovarajući prozor, dakle prozor tipa **Form** za formular, odnosno prozor tipa **Code**·D· za tekst programa. U jednoj aplikaciji može biti i više formulara (ali se naredbom **Run**·D· **Set Startup Form** mora naznačiti koji će biti prvi) i više modula.

Ostatak programa deli se na dve vrste modula: za formulare i za programski tekst. Ove dve vrste modula su međusobom nezavisne: deklaracije iz jednog tipa ne vrede u onom drugom. Obe vrste module imaju istu strukturu: promenljiva deklarisane na nivou modula, i lokalne promenljive, deklarisane po procedurama i funkcijama. Promenljive na nivou modula traju duže od lokalnih, tako da se pomoću njih može prenositi vrednost između potprograma. (Zahvaljujući posebnoj naredbi **static** promenljive u potprogramima mogu trajati duže od čisto lokalnih promenljivih.)

I ostale naredbe bejzika pretrpele su logične izmene. Postoji IF...THEN...ELSE kao i varijacija ELSEIF. Naredba CASE se ovde zove SELECT CASE. DO je sada DO WHILE...LOOP ili DO...LOOP WHILE, a postoje i DO UNTIL...LOOP, odnosno DO...LOOP UNTIL. FOR petlja ostaje ista. Potprogrami se dele na procedure i funkcije, a prenos vrednosti može biti i po vrednosti i po imenu. Sve je to sasvim prepoznatljivo i preuzeto iz drugih modernih jezika: vizuelni bejzik ionako nikom neće biti „maternji" programski jezik.

OBJEKTNI JEZIK?

Svaka druga ili treća reč u vizuelnom bejzik je „objekat". Da li je to onda objektni jezik!? Da bi se odgovorilo na to pitanje, treba se prisetiti koji su to uslovi koje jezik mora da ispuni da

bio objektno-orijentisan: modularnost, apstrakcija podataka, klase, dinamičko zauzimanje i oslobađanje memorije, nasleđivanje, polimorfizam, programiranje pomoću prototipova.

Modularnost ima. Sprovedena je na nivou formulara i programa; promenljive iz tih dveju vrsta modula ne mogu se mešati, pa je moguća i apstrakcija podataka (dakle, ispunjen je i drugi uslov). Treći uslov, mogućnost stvaranja klasa, je delimično ispunjen, i to preko pojma niza upravljačkih objekata. Element takvog niza automatski poprima ponašanje ostalih, što liči na klasičan pojam klase. Element tog niza zadržava sopstveni identitet (osobine mu se mogu menjati nezavisno od drugih) pa je to nešto što liči na nasleđivanje. Formularima, upravljačkim objektima i posebnim objektima mogu se upućivati poruke preko metoda, u odnosu na njih postoji i polimorfnost (isti metod deluje različito na različite objekte). Niz upravljačkih objekata ne može biti višedimenzioni, tako da o višestrukom nasleđivanju nema ni govora. Postoji dinamička alokacija memorije (za nizove) a pošto se svaki niz može redefinisati i da bude dužine nula, postoji i neki primitivni oblik „skupljanja otpadaka". Kažemo „primitivni" jer poenta je baš da se uklanjanje nepotrebnih objekata obavlja automatski, a ne pod svesnom programerovom kontrolom.

Vizuelni bejzik je sav sazdan oko pojma prototipa programa: prvo se pravi korisnički interfejs, pa tek onda reagovanja na sve događaje koje programer želi da dopusti korisniku, kada se sve to ispiše i program je gotov! Završena aplikacija se može prevesti u EXE oblik, pa se tako dobija zaokružena aplikacija koja se drugima može distribuirati (podrazumeva se da oni imaju *Windows 3.0*).

Vizuelni bejzik je i interpreter i prevodilac. Najpreciznije bi bilo reći da se radi o jeziku koji je konglomerat pozitivnih trendova u savremenom programiranju za operativne sisteme sa grafičkim interfejsom. Elementi objektnog programiranja su skoro svi tu – ali samo za isporučene programske komponente. Jezik kojim piše programer je i dalje bejzik, znatno proširen i prilagođen okolini, ali ipak samo „običan" proceduralni jezik. Za programera od objektnih osobina jezika nema ništa. U vizuelnom bejziku je primenjen „princip mekintoš": šta može, može grafički, sa pesmom i veseljem, šta ne može, nikako ne može. Ako programeru jedan od predefinisanih događaja na koje objekti reaguju nije po volji, pa eto, baš šteta, ali uvesti neki novi tip događaja – nemoguće. Pravičnosti radi treba reći da je sa vizuelnim bejzikom moguće skoro sve što treba običnom smrtniku.

Programiranje za *Windows*-e je drastično drugojačije od bilo čega drugog na PC računarima. Za novu sredinu potrebni su i novi jezik i novi pristup. Šansa vizuelnog bejzika je da barem na nivou reklame ponudi „stari jezik" (podrazumeva se da pristup ionako mora biti nov). No, ovaj bejzik je samo po sporednim elementima „stari dobri bejzik". Ne očekujte da za jedan dan počnete da pravite programe sa vizuelnim bejzikom, ali možete biti sigurni da će vam za to trebati mnogo manje vremena nego sa bilo kojim drugim jezikom koji je trenutno na raspolaganju za *Windows*-e.

Korisna adresa

Microsoft GmbH, Edisonstrasse 1
D 8044 Unterschleissheim, Deutchland
Tel. 9949 89 3176 3831
Fax. 9949 89 3176 3600

KAKO OBRAČUNATI PLATU

Više od jedne decenije nakon što je kod nas stigao prvi komercijalni program za personalne računare, i nama je najzad pošlo za rukom da lansiramo softverski paket po svetskim standardima. Program je namenjen poslovima koji se obavljaju u svakoj firmi, obračunu plata, a njegov izdavač je Jugodata iz Beograda.

Na našem napaćenom softverskom tržištu retko se pojavljuju programi koji se prodaju po principima koji su uobičajeni u inostranstvu. Neuporedivo češća situacija je da se program radi po narudžbini jednog ili najviše nekoliko korisnika. Tu i tamo se pojavi neki program koji je usmeren na veće tržište, ali takvi programi u startu imaju ogromne probleme: pirati, velike razlike u potrebama raznih korisnika itd.

Obračun plata je jedan od retkih poslova koji se u svakoj firmi mora raditi. Programa koji taj posao rade na računarima ima puno. Obično su to aplikacije rađene po narudžbini pojedinih firmi, prilagođene su isključivo za tog naručioca.

Nedavno se, međutim, pojavio jedan program koji je izuzetak od ovog pravila. Beogradska firma „Jugodata" izdala je program *Plate* (autor *Boris Terzić*) koji se prodaje baš kao i inostrani programi: priručnik i dve diskete (od 5.25 i 3.5") upakovani su u veoma atraktivnu kutiju, a uz program dobijate i registracionu karticu koju šaljete proizvođaču. Kada jednom kupite program i registrujete ga, sve kasnije verzije dobijate uz povlašćene uslove. Ukoliko ste registrovani korisnik, možete se obratiti na jedan od telefona „Jugodate" za pomoć pri instalaciji i korišćenju programa.

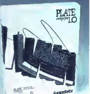
Program se prodaje u dve verzije: verzija 1.0a omogućava obračun plata za firme sa najviše 10 radnika, a verzija 1.0b omogućava obračun u firmama sa više zaposlenih. U vreme pisanja ovog teksta (druga polovina marta) cene ovih programa su bile oko 25.000 (verzija 1.0a) i 50.000 (verzija 1.0b).

Rad sa programom je veoma jednostavan. Na disketi se nalazi program za instalaciju koji od vas traži registracioni broj i naziv firme. Bez obzira na ovu kontrolu, program možete instalirati više puta i tako poštedeti sebe komplikacija sa raznim „zaštitnim" merama. U uputstvu se čak kaže da se prelazak na sledeću godinu obavlja tako što se program ponovo instalira. Sve u svemu, za instalaciju vam neće trebati neko posebno predznanje: najelementarnije poznavanje MS-DOS-a je više nego dovoljno.

Program je, tehnički gledano, korektan, dobar izgled ekrana i jednostavno rukovanje odgovara korisnicima koji nemaju naročitog iskustva. Moglo bi se uputiti i nekoliko zamerki: bilo bi dobro da program podržava neki pregled izveštaja pre štampanja. Doduše, moguće je usmeriti štampanje u datoteku, ali pregled te datoteke morate obaviti van programa, što nije najzgodnije. Takođe, veliki nedostatak je help sistem, naročito kad se uzme u obzir kome je program namenjen. Naime, help sistem radi jedino u osnovnom nivou, a i onda ispisuje samo ime autora i telefon na koji možete tražiti pomoć. Takođe bi bilo dobro kad bi program podržavao malo detaljniju kontrolu štampača.

Kada se program instalira, prvo se unose potrebni podaci o svim radnicima: šifra, ime i prezime, adresa, mesto, broj članova porodice itd. Šifru morate sami odrediti za svakog radnika, program vam tu neće pomoći eventualnim predlogom. Ovo je zgodno ako već koristite neki metod šifriranja, ali u suprotnom može doći do manjih poteškoća. Jednom unete podatke možete kasnije ažurirati bez posebnih ograničenja. Jedino morate paziti da ne duplirate šifre koje su eventualno postojale pa su izbrisane – tako može doći do greške u obračunu.

Najosetljiviji deo posla je sam obračun plata. Program podržava razne varijante obračuna. Najjednostavniji način je da se obračun vrši na osnovu nekih „eksternih" pravila, pri čemu se programu samo saopštavaju gotovi iznosi plata. Osim ovoga (obračun bez bodova), možete izabrati obračun na osnovu „jedinstvenog" (na osnovu broja bodova i procenta realizacije program radi proračun) ili „dvostrukog bod principa" (unose se dve vrednosti bodova za svakog radnika, a zarada se računa kao zbir). Svaki od ovih principa se može kombinovati sa izv. **bruto** i **neto** obračunom – zakonski obračun se primenjuje na bruto ili



neto iznos plate. Sam obračun može biti potpuno automatski, naravno, ako nije potrebno posebno uneti iznos plate za svakog radnika. Po završenom obračunu, podaci za svakog radnika se mogu „ručno" doterati. Program omogućava jednostavno štampanje platnih listića i „prateće" dokumentacije: platnog spiska i obrazaca koje zakon predviđa. Naravno, program omogućava i naizbežni obračun plata u više delova (maksimalno tri).

Razne parametare koji se relativno često menjaju podešavate izborom odgovarajućih opcija: vrednost boda, visina poreskih stopa i slično. Možete isprobavati razne varijante obračuna, preračunavajući neto u bruto iznose i obrnuto, uklapajući tako vašu situaciju u zakon.

Da li će ovaj program odgovarati situaciji u vašoj firmi, zaista je teško reći. Program *Plate* predviđa osnovne izmene zakona i njih lako možete ubaciti u svoj obračun. Međutim, naše firme imaju **veoma** specifična obračuna plata koje je zaista teško podvesti pod neko univerzalno pravilo. Ako vaša firma ima standardni obračun plata i manji broj radnika, ovaj program može biti najjeftinije i najjednostavnije rešenje. Međutim, ako postoje elementi koje program ne sadrži, morate konsultovati izdavača zbog eventualnih izmena ili odustati od kupovine. Takođe, program je napodesan i ako planirate da uvedete računare i u druge vidove poslovanja: ostali programi (drugih autora) neće moći da komuniciraju sa programom *Plate*, što može biti nezgodno. Najpreporučljivije je da pre kupovine konsultujete „Jugodatu" i detaljno se dogovorite. Sve u svemu, konačna odluka je na vama.

Nenad Batocanin

Korisna adresa

Jugodata
11000 Beograd, Bulevar revolucije 326
Tel. (011) 419-326, Fax. (011) 419-768

BRZO KROZ PROZOR!

Pisanje programa za rad pod *Windows*-ima je zaista prava noćna mora. Opremite se standardnim alatom u obliku SDK paketa sa odgovarajućom literaturom i nađete se pred zidom koga je, čak i uz obimnu literaturu i veliki broj programskih primera koje ćete dobiti, zaista teško preskočiti. Zato se pred svaki razvojni alat za *Windows* okruženje odmah postavlja pitanje – može li se njime prekočiti ovaj zid i konačno bez muke ući u jedan prilično zanimljiv svet? U slučaju Microsoft *Quick C za Windows* odgovor je, svakako, ne. Ali, uz njega je bar penjanje lakše.

Zoran Životić

Quick C for Windows 1.0 paket čine pet disketa (1.2 Mb) i četiri knjige. Od *Vodiča kroz razvojno okruženje* (150 str), preko *Alata* (300 str) i *C za Windows* (650 str) do referentnog priručnika za *Windows* programiranje na oko 1.000 strana. Ukoliko ste se već ranije opremili neophodnom majkrosoftovom literaturom, u ovom paketu nećete pronaći mnogo toga novoga. Referentni priručnik deluje potpuno identično ranijem izdanju *Windows Programmers Reference*, dok je *C za Windows* sastavljen iz opšteg uvodnog dela o C programskom jeziku i nešto skraćenog *Windows Guide to Programming*. Priručnik za alate je kraći od ranijeg, jer se u ovom paketu ne nalazi sve što ide uz SDK, ali ćete u njemu naći i opis glavnog novitela – *QuickCase W*.

BORBA ZA PROSTOR

Instalaciju karakterišu, sasvim u skladu sa *Windows* okruženjem, dve stvari – izuzetna jednostavnost i borba sa slobodnim prostorom na disku. Inicijalno je izabrana samo opcija za generisanje *Windows* izvršnih datoteka u *small* memorijskom modelu i za nju je potrebno obezbediti oko 6.5 Mb prostora. Iako je najkrupnija stavka *on line help*, bez koga će biti dovoljno i nešto manje od 4 Mb, nipošto se nemojte lišiti ove opcije. Ukoliko se odlučite i za generisanje DOS programa u sva četiri memorijska modela, onda budite spremni da na disku oslobodite čitavih 14.7 Mb. Za početak vam svakako preporučujem da ipak ubacite generisanje DOS EXE programa, makar samo u *small* modelu – razloge ćete pronaći u ostatku ovog prikaza.

Ozbiljna mana instalacije je nemogućnost razdvajanja memorijskih modela za DOS i *Windows* okruženja – ako vam je, recimo, potreban DOS *large* model, instalacija će generisati biblioteku i za *Windows*, za koji je ona praktično neupotrebna.

Nema razloga da se lišavate bilo koje od ostalih opcija/alata koje instalacija nudi. *QuickCase W* alat za razvoj aplikacija, *dijalog editor* i *editor slića* su svaki za sebe dovoljno vredni da opravdaju neznatno veću količinu zauzetog prostora na disku. Instalaciju možete ponoviti i naknadno, birajući samo one elemente koje želite da dodate postojećoj. Instalacija će biti završena za 5 do 10 minuta u zavisnosti od izabranih biblioteka i u program menadžeru ćete dobiti novu programsku grupu sa četiri ikone.

OKRUŽENJE – EDITOR

Startovanjem *Quick C for Windows* ulazite u okruženje koje je praktično *Windows* implementacija PWB (*Programers Work Bench*) iz MSC 6.0. Ideja koja je u „sešici" doživela poraz zbog očajne implementacije (tromost čak i na najbržim mašinama) pod *Windows*-ima konačno dolazi na svoje. Koliko god imali reči kritike sa DOS varijantu, toliko možemo naći pohvala za ovu verziju. Windows okruženje prevega zbog dinamičkog linkovanja koje omogućava složene programe bez posebnog opterećenja memorije, daje mogućnost ubacivanja velikog broja suptilnih opcija koje olakšavaju rad. *Quick C for Windows* u svemu sledi ovu filozofiju – učitajte odmah bilo koji od C primera iz SOURCE direktorija i, ako imate monitor u boji, odmah će vam pasti u oči jednostavan ali izuzetno praktičan detalj – za razliku od samog teksta programa, koji je ispisan uobičajenim crnim slovima, svi komentari su zelene, a sve službene reči C-a plave boje. Kasnije ćete ustanoviti da različitim bojama možete prikazivati i stringove, *int* i realne brojeve ili potpuno isključiti ovu opciju. Teško je opisati koliko ovakav ispis može da znači za preglednost programa – u svakom slučaju, zbog malog broja službenih reči C-a opcija deluje zaista nenametljivo, a potpuno eliminiše mogućnost da ne uočite otvoren komentar, string ili pogrešno otkucanu reč kao recimo „unsigend".

Ono o čemu uopšte nisam razmišljao pre raspakivanja ovog paketa je da će se u njemu naći rešenje problema programerskog editora pod *Windows*-ima kako bi se konačno oslobodili skromnog NOTEPAD-a. Već nakon nekoliko minuta rada je jasno da se radi o sasvim solidnom editoru. Možete otvoriti više prozora za tekst unutar osnovnog (klasičan MDI – *Multiple Document Interface*), tu je i famozni *tool bar* (panel sa alatima) ispod osnovnog menija sa kontrolama za prevođenje tekućeg programa, obradu kompletnog projekta, dibager itd. i niz mogućnosti za podešavanje editora prema sopstvenim navikama (izbacivanje kontrola za pomeranje teksta, način tabulacije i slično). Editoru nedostaju neka napredne opcije koje srećemo kod DOS programa slične namene – možda bi najprecizniji opis bio da ćete u njemu pronaći minimum opcija, ali još uvek dovoljnih za komforan rad.

S obzirom da je editor deo integrisanog okruženja za razvoj programa, pridodate su mu i opcije za tretiranje niza modula kao skupa koji čini projekat. Otvorite projekat, zatim pritisnete Ctrl+1 i tako jednim potezom učitate sve datoteke u više odvojenih prozora u istom rasporedu u kojem ste ih ostavili kada ste stanje snimili. Pored toga, u menijima se nalaze, kao stavke, poslednja četiri projekta i poslednje četiri datoteka sa kojima ste radili, tako da je dovoljan jedan potez mišem da se one nađu u editoru.

Rad u višeprogramskom okruženju, kada postoji mogućnost da nekim drugim programom izmenite datoteku koju upravo editujete, rešen je zaista originalno. Umesto zaključavanja datoteka u toku editovanja, što je manje-više standardna praksa, *Quick C for Windows* dozvoljava da datoteka nad kojima radite menjate bilo kojim drugim programom. U trenutku kada se vraćate u editor, dobijate upozorenje da je datoteka menjana i mogućnost da automatski preuzmete novu verziju.

Editor je nastao korišćenjem delova koda iz *Microsoft Worda za Windows 2.0* – sličnost je više nego očigledna. Tako se desilo da dominantno mesto (više od pola panela) među kontrolama zauzima deo za izbor fonta – ta kontrola je svakako često korišćena u tekst procesoru kakav je *Word*, ali se zato u programerskom editoru koristi možda jednom u životu – da biste izabrali font sa kojim najprijatnije radite.

Iako nemamo ozbiljnijih zamerki na editor, šteta je što mu nije posvećen još koji trenutak pažnje – to se, pre svega, odnosi na nemogućnost redefinisanja komandnih tastera i nepostojanja makro naredbi.

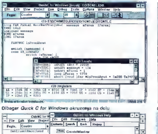
OKRUŽENJE – ALATI

Alati čine poseban deo priče i sa njima se svakako vredi odmah upoznati. Razvoj programa počinjete korišćenjem najvažnijeg od njih – *QuickCase W*. Radi se o alatu za dizajniranje osnovne strukture programa i *Windows* interfejsa. Rad se svodi na to da u osnovni prozor vaše aplikacije, koji za vas otvori *Case W*, ubacujete i izbacujete stavke iz menija i to do tri nivoa – pored svake stavke padajućeg menija možete ubaciti i još jedan nivo *kaskadnog* menija. Svaku stavku možete povezati sa dijalogom koji ste kreirali u dijalog editoru, a možete u program ubaciti i kod za obradu dijaloga koji nisu direktno vezani sa menijem.

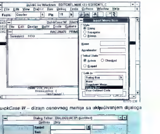
Kada završite posao, *Case W* će generisati sve potrebne datoteke – od *.mak* datoteke (*makefile*) preko *.def* i *.rc* do C programa i .H zaglavlja. Ostaje vam samo da se vratite u *Quick C for Windows*, otvorite upravo generisan projekat i startujete izradu projekta. S obzirom da će prevođenje proći sigurno bez greška, odmah tasterom F5 možete startovati novi program i uživati u vašem prvom *Windows* programu, koji ste generisali bez i malo truda ili poznavanja *Windows* programiranja.

Naravno, iskusni *Windows* programeri sigurno neće biti toliko oduševljeni, ali će svakako biti zadovoljni. Program zapravo rešava pitanje prvog koraka u izradi programa, koji jeste mali, ali je tehnički gledano najgori i najdosadniji. *Case W* generiše *WinMain* funkciju sa pozivom za registrovanje klase osnovnog prozora, kreiranje prozora i centralnu petlju za obradu poruka. Istovremeno generiše i funkciju za obradu poruka osnovne klase, u kojoj na klasičan C *switch* način, putem niza *case* situacija, ostavlja prostor za ubacivanje koda za obradu pojedinih komandi iz menija. Pored toga, u C modulu će se naći i sve funkcije za obradu dijaloga koje ste povezali u program sa delom za poziv dijaloga ukoliko je on vezan za neku opciju iz menija. Naravno, dijalog funkcija sadrži samo spisak slučajeva koje treba obraditi – na vama je da inicijalizujete dijalog i ugradite sve ostale potrebne delove koda. I tu je kraj mogućnosti *QuickCase W*.

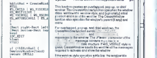
Generisani program može biti iskomentarisan do nekoliko nova – najviši nivo ubacuje čitava ekrane opisa sa prilično opštim napome



Dibager *Quick C for Windows* okruženja na delu



QuickCase W – dizajn osnovnog menija sa uključivanjem dijaloga



Izuzetno obiman i posebno vredan HELP



Redizajnirani dijalog editor

nama iz kojih sigurno nećete naučiti da programirate pod *Windows* om tako da ćete se posle zadovoljenja početne radoznalosti sigurno odlučiti za manje ali daleko značajnije komentara uz svaki *case* kojima se referiše na tekst stavke iz menija. Tako se dobija pregledan program u kome je lako pronaći mesto na kome treba dodati kod za obradu svake od stavki menija.

Case W je pored generisanja programa u stanju da izvede i daleko delikatniju operaciju izmena na već delimično menjanom kodu. Naime, ako u toku razvoja aplikacije odlučite da treba izmeniti nešto u osnovnom dizajnu aplikacije i to uradite uz pomoć ovog alata, *update* opcija će učiniti sve da u izvorni program unese samo nastale izmene ne dirajući u deo koda koji ste vi dopisali. Već na prvi pogled je jasno da ovakva opcija može biti izvor problema, pa treba ipak voditi računa da se generisani deo koda što manje menja – najjednostavnije je kod za obradu pojedinih stavki realizovati kao posebne funkcije, a samo pozive ubaciti u odgovarajući *case*.

Testirati do kraja ovu opciju je prosto nemoguće – jednom mi se desilo da potpuno izgubim sadržaj generisane funkcije koju sam u međuvremenu prilično izmenio – *Case W* je praktično celu funkciju generisao ponovo potpuno uništivši moje izmene. Ipak, slučaj se desio u samom početku rada, dok još nisam upoznao alat niti sam precizno vodio evidenciju o tome koje sam izmene napravio – s obzirom na kasnija iskustva sa ovom opcijom koja su prolazila bez ikakvih problema, moguće je da sam prevideo neku krupnu izmenu koja je je dno takvim efektom i mogla rezultirati.

Iz priloženog prospekta i poziva na doplatu za dve odvojene verzije *Case W* saznajemo da se radi se o proizvodu nezavisnog proizvođača CASEWORKS. Uz doplatu od 99 USD dobija se *Case W Standard Edition* koji nudi dva bitna poboljšanje – u stanju je da odmah animira izvršavanje aplikacije (menija i dijaloga) i da generiše kod za povezivanje dijaloga sa programskim promenljivima. Za doplatu od 249 USD dobija se *Corporate Edition* koji ima niz dodatnih opcija, među kojima su generisanja MDI interfejsa, interfejsa prema standardnom *Windows* help mehanizmu i obradu nezavisnih ikona i kontrola unutar klijent područja prozora.

Ostala dva alata – dijalog editor i editor slika (ikona, kursora i bit mapa) su samo nove verzije već poznatih programa. Dijalog editor je nešto drugačije dizajniran, dok se editor slika, sada nazvan *Image editor*, skoro i ne razlikuje od ranije verzije *SDK Paint* programa. Od ostalih alata koji su uključeni u SDK najviše nedostaje izuzetno konstan SPY – šteta je što celina nije zaokružena i ovim malim programom.

Svi alati se nalaze kao nezavisne ikone u grupi *QC/Win* koja se kreira instalacijom. Alati su i pored toga postavljeni u osnovni meni glavnog programa, odakle ih je jednostavnija pozivati, jer im integrisano okruženje prenosi direktorij u kome se projekat nalazi, tako da se potrebne datoteke lako pronalaze. U meni sa alatima mogu se dodati i drugi programi koji nemaju veze sa *Quick C for Windows* okruženjem.

KOMPAJLER I LINKER

Kompajler i linker su već poznate verzije *MS Quick C* kompajlera i *Segmented Executable Linker 5.10*, ali sada prepisana specijalno za *Windows* okruženje (qc.exe je postao qcw.exe, link.exe je postao linkw.exe itd). Pored generisanja DOS i WIN EXE programa, moguće je direktno praviti i DLL (*Dynamic Link Libraries*) biblioteke, ali i *QuickWin EXE* koji je neka mešavina DOS-a i *Windows*a – svaki klasično napisan DOS program se može prevesti u ovom formatu bez ikakvih izmena, a rezultat startovanja ovakve aplikacije je prozor u kome se prati ispis usmeren na *stdio*. Domet ovog formata je vrlo ograničen – ideja je da se običnim DOS programima (koji moraju biti napisani bez ikakvih direktnih gađanja video memorije i sličnih „optimizacionih" tehnika – ima li takvih?) jednostavno da *windows look*.

Jedini vidljiv rezultat prepisavanja kompajlera i linkera za *Windows* okruženje je izveštaj o toku prevođenja – na ekranu se pojavljuje dijalog u kome se prikazuje broj obrađenih linija, a samo jedno dugme *Cancel* omogućava da proces u bilo kom trenutku prekinemo. Naravno, dok traje prevođenje možemo preći u bilo koji drugi *windows task*, ali ne i nazad u sam *Quick C for Windows*.

Verzija qcw kompajlera je 2.60 što nam na govori mnogo, jer je ova varijanta u MSC 6.00 bila sakrivena samo kao opcija qc osnovnog kompajlera. Po tome što prepoznaje *_asm* direktivu, reklo bi se da se radi o istoj ili novijoj verziji koja je ugrađena u C 6.00. Ipak, pokušaj

da se program prevede u oba okružnja pod potpuno istim uslovima nije dao identičan rezultat. Očigledna razlika je bila oko 3% duži EXE koji je generisao *Quick C for Windows*. U svakom slučaju ovaj kompajler omogućava brzo generisanje i DOS programa bez ograničenja u pogledu mogućnosti. Iako da nam uvek na kraju ostaje da finalnu verziju prekompajliramo sa MSC 6.00 i tako dobijamo najkvalitetniji kod.

Brzina samog prevođenja u .obj sa uključenom optimizacijom je po mojim merenjima negde oko 40 000 linija u uslovima kada je *Quick C for Windows* jedini aktivni program na 33 MHz 386 računaru sa 4 MB memorije od kojih je 1 MB odvojen za disk keš program (HYPERDISK). Iako ova cifra deluje izvanredno, radi se o znatno manjoj brzini prevođenja u odnosu na DOS verziju (MSC 6.00 puna verzija deklarisana je na 60 000 linija/minut, mada je realno negde oko 60 000) i podatku koji ipak zavarava – vaš program od recimo 4000 linija neće biti preveden za 6 sekundi. Naime, u broj linija su uračunata i zaglavlja, što je svakako korektno, ali realno gledano to znači da čak i najmanji progam od recimo 500 linija biva preveden za 6-7 sekundi, jer je u njega obavezno uključen i *windows.h* koji ima oko 3 500 linija. Bez obzira na brzinu samog prevođenja, ukupno vreme generisanja koda nikada nije manje od 30-tak sekundi. 10-15 sekundi je potrebno dok se uopšte pokrene kompajler i još toliko za fazu povezivanja koje opet, iako se sam proces izuzetno brzo izvrši, zahteva dodatno vreme dok se pokrene linker.

Način na koji se konfigurišu parametri prevođenja je poznat iz PWB-a. Putem dijaloga biraju se opcije kao što je *Optimizacija Da/Ne/Puna* ili *Uključivanje provere steka Da/Ne*, a program ih prevodi u konkretne prekidače koje prenosi u komandnoj liniji pri pozivu kompajlera. Pri tome se održavaju dva seta opcija – jedan je posvećen generisanju radne verzije programa u koju je uključena podrška dibagera, a drugi za finalnu verziju. Pri generisanju radne verzije programa moguće je uključiti i interesantnu opciju koja je do sada bila poznata samo kod linkovanja, takozvani *incremental compile*. Kompajler je u stanju da na nivou funkcije detektuje da u njoj nije vršena nikakva izmena i da funkciju ne prevodi ponovo, već iskoristi ranije generisan kod. Opcija kod jako dugačkih programa može drastično da skrati vreme prevođenja, ali je realno ipak od male koristi. *Windows* programi se inače pišu razbijeno na sitne module, tako da ova opcija nema mnogo šansi da se dokaže. Eventualna upozorenja koja je kompajler izdao i koja nisu odmah raščišćena u sledećem prevođenju neće biti prikazana jer program i ne dira funkcije koje nisu menjane, tako da opcija može kod nepažljivog korišćenja izazvati i probleme.

Prijavljivanje grešaka je takođe izvedeno na već standardan način za ovakva okruženja. Program otvara poseban prozor i u njega upisuje poruke kao tekst *samo za čitanje* tipa. Postavljanjem kursora na neku od poruka i pritiskom na taster *Enter* prelazimo u prozor sa modulom u kome se greška nalazi, a kursor se automatski postavlja na odgovarajuću liniju.

POMOĆ NA SVE STRANE

Među najprijatnije delove ovog paketa spada dibager. Sam dibager je sasvim klasičan i već viđen u integrisanim okruženjima za razvoj, ali je njegovo prisustvo značajno zato što se konačno i testiranje *Windows* aplikacija može obaviti jednostavno. Tu su sva uobičajene funkcije – izvođenja programa korak po korak, postavljanje prekidnih tačaka, praćenje sadržaja registara i promenljivih itd. Uzrok čuvenog *Trap #13* koji označava zalazak programa u nedozvoljena memorijska područja, sada se lako uočava jer dibager staje na liniji koja je uzročnik. Naravno, često su uzrok loše preneti parametri nekoj od sistemskih funkcija i tada dibager ne može da ukaže na liniju sa greškom jer ona i nije u našem programu, ali se ponovnim izvođenjem programa korak po korak lako pronalazi poziv funkcije koji je sporan. Dibager je opremljen i posebno korisnom *quick watch* funkcijom – pritiskom na *Shift F9* (ili izborom dugmeta iz panela sa alatima) pojavljuje se prozor sa sadržajem promenljive na kojoj se trenutno nalazi kursor.

Poslednje, ali svakako takođe prijatno iznenađenje je i ugrađeni *HELP* koji sve referentne priručnike čini potpuno suvišnim. Sreća je da je tako i da svoje interesovanje možete usresrediti samo na knjigu *C za Windows* koja će vas uvesti u tajne programiranja za *Windows* okruženje – listanje referentnog priručnika sa spiskom funkcija, struktura i poruka na 1 000 strana nije ni malo praktično. Dizajn HELP-a je prilično izmenjen, a i unutrašnja struktura programa – sa starim helpom ne može se koristi novi sadržaj, ali novi help prihvata i starije datoteke. Help deluje na samo po mom sudu manje atraktivno nego stari, ali je po funkcionalnosti ipak bolji. Ono što se iz instalacije ne vidi (mada je naznačeno u dokumentaciji) je da će novi help prebrisati stari, tako da ga treba pre instalacije sačuvati ako vam se ovaj ne dopadne. Ipak, verujem da se samo zarad estetske vrednosti ne isplati ostati bez novog helpa, jer je u njemu, za razliku od SDK helpa, objedinjeno sve – i *Windows* funkcije i C funkcije. Praktično su spojena dva odlična helpa – onaj iz MSC 6.00 i SDK paketa, čime ovaj deo *Quick C for Windows* paketu daje posebnu vrednost. Help se može koristiti potpuno kao nezavisna aplikacija, ali je i integrisan sa ostatkom paketa – nakon tastera F1 prelazi se automatski u help na onu odrednicu na kojoj se kursor nalazio u editoru.

IZUZETNA OPERATIVNOST

Zaključani sud o *Quick C for Windows* verzije 1.0 se može svesti u samo dve reči – izuzetna operativnost. U paketu nema skoro ničeg novog u odnosu na ono što smo već videli pod DOS-om, ali je *Windows* okruženje iskorišćeno na pravi način i programerima je ponuđen alat kojim se aplikacije mogu razvijati izuzetno jednostavno, uz svu neophodnu pomoć u radu. Paket će pristalicama *Windows* okruženja značiti mnogo više nego sama činjenica da je namenjen razvoju *Windows* aplikacija. Čak mi se čini da je ta namena sasvim u senci onoga što se generalno dobija jednim ovakvim okruženjem za razvoj koje je univerzalno primenjivo. Zato nisam ni osetio potrebu da se posebno bavim stvarima koje su inače presudne za kompajler, kao što su pouzdanost i kvalitet izlaznog koda. *Quick C for Windows* jednostavno ne treba tretirati sa tog stanovišta. Kvalitet izlaznog koda i pouzdanost su svakako dovoljni (bar po dosadašnjim iskustvima, mada takav sud ne mogu konačno potvrditi bez više iskustva) za fazu razvoja. Za finalno kompajliranje neke komercijalne aplikacije uvek nam je na raspolaganju standardni C 6.00.

Korisna adresa

Microsoft GmbH, Edisonstrasse 1
D-8044 Unterschleissheim, Deutschland
Tel. 9949 89 3176-3831
Fax 9949 89 3176-3800

Naučno-stručni skup na sajmu TEHNOTRONIKA

INFORMATIKA 92.

U okviru ovogodišnje računarske izložbe "Tehnotronika" u Beogradu biće održan i zanimljiv naučno-stručni skup "Informatika 92". Organizatori skupa su Društvo za informatiku Srbije i Beogradski sajam, a sponzori DIGIT – Računarski inženjering i časopis RAČUNARI.

"Informatika 92" će se održati 14. maja 1992. na Beogradskom sajmu, u velikoj sali hale XIV, sa početkom u 10.00 sati.

Za skup je predviđen sledeći

PROGRAM

Nikola Marković: Otvaranje skupa
Dr Jozo Dujmović: Neuronske mreže i njihova primene
Dr Jovan Đorđević: Arhitektura i organizacija računara upravljanih podacima
Gojko Ribar: Case alat – prototipski razvoj softvera
Dr Slobodan Krčevinac: Modeliranje vrednovanja imovine preduzeća
Ing. Dragan Grebović: Metoda projektovanja sistema zaštite u mrežama računara
Nikola Marković: Zaštita podataka i softvera
Dejan Ristanović: Tendencije u razvoju PC tehnologije
Dr Ljiljana Milić: Razvoj modernih telekomunikacija u svetu i kod nas
Dr Vilman Gajović: Informacioni projekti u telekomunikacijama
Dušan Tatomirović: Modemske komunikacije i obrazovanje
Dr Dušan Starčević, dr Čeda Milenković: Digitalizacija slika
Predrag Milićević: Ćirilični set znakova u informatici
Darko Kovač: Upravljanja resursima u BEKO-u

Prijava i uplata kotizacije od 4 000 dinara vrši se do 30. aprila 1992. godine na žiro račun naveden u prijavi (posle tog roka kotizacija iznosi 6 000 dinara, a uplaćuje se ili na žiro račun ili na samom skupu). Učesnici će na početku skupa dobiti zbornik referata. Očekuje se da će skup trajati do oko 17.00 sati.

Informacije o skupu se mogu dobiti na telefon 011/ 602-266.

DRUŠTVO ZA INFORMATIKU SRBIJE, 11000 Beograd, Kneza Miloša 9/IV

P R I J A V A

Prijavljujemo se za učešće na naučno-stručnom skupu Informatika 92, Beograd, 14.05.92.

Ime i prezime

Preduzeće

Adresa

Telefon   Telefax

Kotizaciju za učešće u iznosu od 4 000.00 ili 6 000.00 dinara posle 30.04.92 uplatićemo na žiro račun 60803-678-18621 Društvo za informatiku Srbije

U   Datum

Potpis

Napomena: Može se koristiti i foto kopija ove prijave

KA GRANICAMA MATEMATIKE

Kad god se prisetim svih onih sati koje sam utrošio pokušavajući da smislim optimalan način za baratanje simboličkim izrazima učini mi se da sam tapkao u mestu. U tom pogledu najviše je obećavao sada već penzionisani *Reduce*, ali me je prosto izluđivao svojim kapricioznim izletanjem u *DOS* i nepredvidljivim otkazivanjem poslušnosti. Samostalni napori usmereni na *LISP* i *C++* davali su opipljive ali, ipak, amaterske rezultate. A onda se pojavila *Mathematica 386*, koja me je opčinila svojom konzistentnošću, kapacitetom i efektivnom snagom. Tako sam, po ko zna koji put, iskoračio sa staze kojom sam želeo da prođem sam i krenuo tuđim stopama.

Saša Marković

PRVE IMPRESIJE

Remek delo Stivena Volframa *(Stephen Wolfram) Mathematica 386* koncipirano je kao interpreter i po tome (i samo po tome) je nalik bejziku i sličnim igračkama. Jednostavno od vas se očekuje da na zahtev (**In**) unesete određenu instrukciju da biste odmah zatim dobili odgovor (**Out**). Treba istaći da je *Mathematica 386* potpuno simbolički orijentisana i da jednako efikasno barata kako numeričkim tako i simboličkim (analitičkim) izrazima. Stvar radi tako dobro da ćete već posle nekoliko minuta prestati da uporno proveravate svaki rezultat koji se pojavi na ekranu. Evo nekoliko načina da impresionirate radoznalca koji vam gleda preko ramena. Sa **f [x_] := x Sin [x]** definisite funkciju **f(x) = x sin (x)**. Zatim sa **D [f [x], x]** (ili **f' [x]**) i **D [f [x], {x, 2}]** (ili **f'' [x]**) odredite njen prvi i drugi izvod. Neodređeni integral **int f(x) dx** dobija se sa **Integrate [f [x], x]**, a određeni **int (from pi to t) f(x) dx** sa **Integrate [f [x], {x, Pi, t}]**. Sumu kvadrata prvih pedeset prirodnih brojeva izračunaćete sa **Sum [i^2, {i, 1, 50}]**, a proizvod prvih deset prostih brojeva sa **Product [Prime [i], {i, 1, 10}]**. Sistem jednačina **x + y = a, x^2 + y^2 = 20** rešava po **x**, **y** instrukcija **Solve [{x + y == 6, x^2 + y^2 == 20}, {x, y}]**. Tejlorov razvoj funkcije **f(x)** u okolini tačke **x = 1** do polinoma sedmog stepena može se odrediti sa **Series [f [x], {x, 1, 7}]**, a granična vrednost niza **(1 – 1/n)^n** kada **n –> inf** sa **Limit [(1 – 1/n)^n, n –> Infinity]**. Dijagram funkcije **f(x)** u intervalu (0,10) prikažite naredbom **Plot [f [x], {x, 0, 10}]**. Na kraju, ono što najviše impresionira svakog početnika: broj π na 100 decimala dobija se instrukcijom **N [Pi, 100]**. Ima toga još, ali je *Mathematica 386* ipak mnogo više od pukih ilustracija, makako one efektne bile.

OSNOVNI KONCEPT

Svi objekti (simboli, konstante, formule) u *Mathematica 386* se mogu grupisati u liste nalik onima u *LISP*-u. *Mathematica 386* ima veoma bogat i raznovrstan set instrukcija za manipulaciju listama. Lista formalno nastaje grupisanjem objekata unutar vitičastih zagrada ili eksplicitnom upotrebom instrukcije **List [lista objekata]**. Po svojoj matematičkoj suštini liste su vektori. Analogno, lista čiji su elementi takođe liste predstavlja dvodimenzionalni niz ili matricu. Tako je {**a**, **b**, **c**} vektor dužine 3, a {{**a**, **b**}, {**c**, **d**}, {**e**, **f**}} matrica formata 3 x 2. Za generisanje vektora i matrica najčešće se koristi naredba **Table**. Ako je, na primer, potrebno generisati matricu **a** formata 10 x 5, u kojoj je svaki element jednak zbiru rednog broja vrste i kolone u kojoj se nalazi, upotrebljavamo

a = Table [i + j, {i, 5}, {j, 10}]

Elementima liste (vektora, matrice) pristupa se kao u *C*-u, upotrebom uglastih zagrada. Prvu vrstu matrice **a** dobićemo sa **a [[1]]**, a element a_{23} sa **a [[2]] [[3]]** ili alternativno sa **a [[2 3]]**. Za rad sa matricama stoje na raspolaganju brojne „očekivane" funkcije (matrično množenje, inverzija, transponovanje, izračunavanje determinante), ali su tu i mnogo ekskluzivnije procedure (singularna dekompozicija, izračunavanje karakterističnih vrednosti i slično). Međutim, smisao listi u *Mathematica 386* je mnogo dublji. Interno se svi izrazi, formule i procedure predstavljaju kao liste. Tako je suma **a + b + 2** interno prikazana kao **Plus [a, b, 2]**, odnosno **{Plus, a, b, 2}**, gde je prvi član liste indeksiran nulom. Ovako je omogućeno efikasno manipulisanje smislom izraza jednostavnom zamenom njegovog zaglavlja (proverite dejstvo instrukcija **Apply [Times, x + y + z]** i **Apply [Plus, {a, b, c}]**).

FUNKCIJE I TRANSFORMACIJE

Najmoćnije oružje ugrađeno u *Mathematica 386* je fleksibilan i inteligentan način definisanja funkcija i njihove upotrebe. Neka je potrebno definisati funkciju **f(x, y)** na sledeći način:

$$f(x, y) = \begin{cases} x + y & \text{za } x < y \\ xy & \text{za } x = y \\ x - y & \text{za } x > y \end{cases}$$

Definicija funkcije u *Mathematica 386* maksimalno prati analitički zapis:

```
f [ x_, y_ ] := x + y /; x<y
f [ x_, y_ ] := x y   /; x==y
f [ x_, y_ ] := x - y /; x>y
```

Uočimo da se formalni argumenti funkcije postavljaju u uglaste zagrade i da se iza svakog formalnog argumenta nalazi „_" (*underscore*). Ako ga zaboravite, funkcija će kasnije prihvatati kao svoje argumente isključivo simbole **x** i **y**. Crtica iza imena formalnog argumenta obezbeđuje njegovu proizvoljnost i ne pojavljuje se na desnoj strani definicije. Znak za definiciju funkcije je :=, a uslov definicija navode se iza /;, pri čemu se mogu koristiti i logički operatori identični onim u *C*-u (&& i ||). Dodatno se može zahtevati da se prilikom poziva funkcija proverava tip argumenata i njihov broj, pri čemu se izostavljeni argumenti mogu nadomestiti podrazumevanim vrednostima. Tako funkcija

f [x_Real?Positive, y_:1] := x + y^2

ima dva argumenta: prvi mora da bude pozitivan realan broj, a drugi se može izostaviti, pri čemu se podrazumeva vrednost 1. U ovom segmentu *Mathematica 386* najviše podseća na *C++*.

Rekurzije su sasvim normalna stvar. Večni primer Fibonačijevih brojeva se ni ovde ne može izbeći. Nakon definicije

```
fib [ n_Integer ] := fib [ n-1 ] + fib [ n-2 ]
fib [ 0 ] = fib [ 1 ] = 1
```

Mathematica 386 će sa lakoćom izračunati **fib [10]**. Međutim, mogućnosti *Mathematica 386* su i ovde daleko veće. Kao formalni argumenti funkcije mogu se pojaviti izrazi proizvoljne strukture. Tako će funkcija

f [x_ + y_] := x y

pretvoriti zbir dva simbola u njihov proizvod. Na ovaj način se mogu definisati novi operatori krajnje egzotičnog dejstva. Evo kako je u nekoliko redova moguće koncipirati operator koji će sa lakoćom diferencirati svaki polinom:

```
dif [ c_, x_ ] := 0 /; FreeQ [ c,x ]
(c' = 0 ako c ne zavisi od x)
dif [ x_, x_ ] := 1 (x' = 1)
dif [ c_. x_^n_., x_ ] := c n x^(n-1) /; FreeQ [ {c,n}, x ] ((cx^n)' = cnx^(n-1))
dif [ u_ + v_, x_ ] := dif [ u,x ] + dif [ v,x ]
((u + v)' = u' + v')
```

Probajte sada **dif [3x^2 + 5x – 6, x]**. Nije dakle za čuđenje što se uz pojam *funkcije* u priručniku jednako često koristi pojam *transformacije*.

Radeći (ili se zabavljajući) sa *Mathematica 386* mnogo puta ćete doći u situaciju da vrednost nekog izraza treba izračunati za određenu vrednost njegovih argumenata. Na primer, neka ste tokom sesije definisali simbol **t** kao **t = x^3 + x + 1** i želite da njegovu vrednost izračunate za **x = 3**. Ništa prostije, kažete: otkucam **x = 3** a zatim samo **t** i dobijam rezultat. Na ovaj način se, međutim, gubi simbolička struktura simbola **t** koji je sada automatski preračunat za **x = 3** i ima fiksnu vrednost. Tako se privremeno gubi mogućnost da se sa simbolom **t** dalje manipuliše na simbolički način. Da bi se stara forma izraza vratila, mora se simbol **x** „ratosiljati" pridružene vrednosti naredbom **x=.** Da bi se ova komplikacija izbegla (a koja, uzgred budi rečeno, uopšte ne postoji kod „klasičnih" programskih jezika koji barataju samo brojkama) smišljeno je nešto veoma praktično: vrednost izraza može se izračunati i *privremenom* supstitucijom simbola **x** sa 3 svuda gde se on pojavljuje u izrazu **t**:

t /. x –> 3

Nakon ovog izračunavanja simbol **x** i dalje ima vrednost koju je posedovao pre izračunavanja izraza **t**. Ova „pravila" (*rules*) izračunavanja izraza mogu biti komplikovana

Log [x] + x Sin [y] /. {x –> 2.0, y –> 0.5}

i veoma komplikovana

g [5] //. {g [0] –> 1, g [n_] –> n g [n-1]}

Listing 1

```
rksist[f_List, vars_List, vars0_List, h_?NumberQ, end_?NumberQ] =
NestList[rkstep[f, vars, #1, h]&, vars0, Round[(end   vars0[[1]])/h]]
rkstep[f_List, vars_List, vars0_List, h_?NumberQ] - Block[{k1, k2, k3, k4,
d},
k1 = h * f   /    Inner[Rule, vars, vars0, List],
k2 = h * f   /    Inner[Rule, vars, vars0 + Flatten[{h, k1}]/2, List],
k3 = h * f   /    Inner[Rule, vars, vars0 + Flatten[{h, k2}]/2, List],
k4 = h * f   /.   Inner[Rule, vars, vars0 + Flatten[{h, k3}], List],
d = Flatten[{h, (k1 + 2 * k2 + 2 * k3 + k4)/6}], Return[vars0 + d]; ]
rkplot[data_List, n1_Integer, n2_Integer] :=
ListPlot[({#1[[n1]], #1[[n2]]} & )  /@  data, PlotJoined  -> True]
```

Program koji rešava *sistem* **diferencijalnih jednačina prvog reda metodom** *Runge-Kutta* **četvrtog reda**

U ovom poslednjem primeru definisana su dva pravila koja asociraju na definiciju faktorijela. Znak // umesto / obavezuje interpreter da uzastopno primenjuje navedena pravila sve dok se njima modificira izraz na levoj strani. Na ovome se zasnivaju brojne metode za uproščavanje analitičkih izraza. Delimično razočaravajuća činjenica da *Mathematica 386* ne zna da je $\sin^2 x + \cos^2 x = 1$ lako se „ispegla" upotrebom pravila

a_ Sin [x_] ^2 + a_ Cos [x_] ^2 - > a

MATHEMATICA 386 ZA PROGRAMERE

Ma koliko komfora osećali u manipulaciji listama, funkcijama, pravilima i sličnim objektima, bez pravih kontrolnih struktura nema pravog programiranja. Oni koji dobro poznaju C neće imati mnogo toga da nauče. U ovom domenu *Mathematica 386* i C liče kao jaje jajetu, ako izuzmemo donekle istumbane interpunkcijske znake „," i „;".

U komplikovanim definicijama funkcija pojavljuje se potreba za većim brojem lokalnih pomoćnih promenljivih čija imena mogu doći u konflikt sa već definisanim simbolima. U tu svrhu stoji na raspolaganju instrukcija **Block [{x,y, ...}, f]**, kojom se simboli **x** i **y** definišu kao lokalni za proceduru definisanu nizom instrukcija **f**. Pojedinačne instrukcije unutar niza **f** treba odvajati pomoću „;". Evo značajnijih kontrolnih struktura:

- **Do [f, {i,imin,imax,istep}]** izračunava **f** preko **i** koje se menja od **imin** do **imax** sa korakom **istep**. **While [test,f]** izračunava **f** sve dok je **test == True**.
- **For [start,test,step,f]** predstavlja analogno ostvarenje klasične **for**-petlje u C-u. **If [test,f1,f2,f3]** daje **f1** ako je **test == True**, **f2** ako je **test == False** ili **f3** ako se rezultat **test**-a ne može utvrditi usled njegove simboličke prirode.
- **Which [test1,f1,test2,f2, ...]** izračunava **test1**, **test2** ... i vraća vrednost **fi** koja se nalazi iza prvog **test**-a koji daje logičku vrednost **True**.

Potpuno nalik C-u implementirane su i strukture **Switch**, **Break**, **Continue** i **Return** sa potpuno istovetnim akcijama koje izazivaju. Prikažimo kako to izgleda „u komadu" na primeru funkcije koja računa zbir cifara nekog prirodnog broja:

ZbirCifara [n_Integer?Positive] := Block
[
{suma = 0, broj = n},
While [broj > 0,
suma += Mod [broj, 10] ;
broj = Floor [broj / 10] ;
] ;
Return [suma] ;
]

Šta će dati

ZbirCifara [ZbirCifara [ZbirCifara [1992^1992]]] ?

Za razliku od prevodioca kod kojih se argumenti potprograma i funkcija prenose preko steka (vrednošću ili ukazateljem – adresom), prenos argumenata funkcija u *Mathematica 386* vrši se prostim izračunavanjem argumenata i njihovom zamenom u telo funkcije na onim mestima gde u definiciji funkcije figurišu formalni argumenti (supstituciona semantika). Supstitucija je kompletna i bezuslovna: vrši se na obe strane jednakosti, na svakom mestu unutar definicije funkcije. Ako je definisana funkcija **h [x_] := Sin [x] + Log [x]**, tada je postupak izračunavanja **h [0.5]** isti kao prilikom izračunavanja vrednosti izraza

Sin [x] + Log [x] /. x - >0.5

Ovo ima brojne reperkusije na način kojim se u funkcijama sme manipulisati formalnim argumentima.

DA SE NE PROPUSTI...

Pomenimo još neke lepe stvari:

- *Mathematica 386* radi kompletnu grafičku prezentaciju ulaznih podataka u *PostScript*-u. Tako dobijeni dijagrami se mogu spremiti na disk i po potrebi dorađivati i popravljati.
- Rezultati koje *Mathematica 386* proizvodi mogu se izvoziti u datoteke prema *C*, *fortran* ili *TEX* standardu uz minimalne modifikacije.
- *Mathematica 386* raspolaže nebrojenim specijalnim funkcijama – teško da će vam ikad koja zatrebati!
- Program raspolaže ne samo kompletnim setom sofisticiranih matematičkih alatki već i prilično detaljnim skupom naredbi za manipulisanje datotekama, periferijama i rekonfiguraciju sistema. Po potrebi se može aktivirati bilo koja *DOS* komanda – zgodno ako vam zatreba omiljeni editor. Komandnu liniju koju želite da prosledite operativnom sistemu započnite uskličnikom.
- Preciznost numeričkih izračunavanja zavisi samo od vaše želje. Broj značajnih cifara sa kojima *Mathematica 386* može da manipuliše definitivno prevazilazi potrebe i najzahtevnijeg korisnika.

Ipak, ima i nedostataka koji ponekad jako zasmetaju:

- Korisnički interfejs za *DOS*-bazirane sisteme je dosta neugledan i izuzev *PostScript*-grafike ograničen je na tekst mod. Kucanje komande duže od jednog reda i njeno naknadno ispravljanje ponekad prerasta u pravu moru. Nekakvo *Windows* ili *GEM*-okruženje u narednoj verziji bi me obradovalo.
- Da bi stvari radila kako treba, neophodna je 386/387 kombinacija. ATeisti (vlasnici *AT*-a) i oni koji još uvek emuliraju koprocesor nemaju tu šta da traže.
- Uz puno razumevanje detaljno opisanih razloga koji su S. Volframa naveli da ceo sistem zamisli kao interpreter, ipak smatram da nedostaje mogućnost kompilacije gotovih programa. Iako u simboličkom programiranju brzina nije opredeljujući faktor, bilo bi lepo da korisnički definisane funkcije rade barem upola tako efikasno kao ugrađene ultra brze rutine. Na žalost, ovako se sve vreme osećate kao da sedite na zadnjem sedištu najnovijeg mercedesa. Lepo je, nije da nije, ali bi bilo lepše da ste za volanom.
- *Help*-informacije dobijaju se tako što se ispred imena funkcije stavi upitnik „?" uz dozvoljenu upotrebu džoker znakova. Pomoć se obično iscrpljuje u dve ili tri kriptične rečenice, pa na kraju ostaje samo kupusanje po masivnom priručniku.

KROZ OTVOREN PROZOR

Ipak, ukupan utisak je više nego povoljan. Jednostavno, uz malo vežbe počnete da osećate kako se pred vama otvara još jedan prostor koji do sada uopšte niste istraživali jer niste ni imali odgovarajuće instrumente. Aparatura je moćna, poslušna i predvidiva, a pouzdanost vrlo visoka. Ugrađeni interpreter omogućava pisanje preglednih (možda malo uvrnutih) programa za koje je ranije trebalo ispisati stotine linija koda. Ovu činjenicu ubedljivo demonstrira programčić koji prilažem, rešavajući u svega deset linija *sistem* diferencijalnih jednačina prvog reda metodom *Runge-Kutta* četvrtog reda, obezbeđujući uz to i funkciju za grafički prikaz dobijenog numeričkog rešenja. Do sada nisam sreo interpreter ili prevodilac koji omogućava tako nešto na manjem prostoru. Stara je istina da se najjači otrovi drže u malim bočicama.

Neka treba rešiti diferencijalnu jednačinu drugog reda **y'' + 1/2 y' − y = 0** gde je **y' = dy/dx** i **y(0) = y'(0) = 1**. Nakon smene **z = y'** diferencijalna jednačina se raspada na dve diferencijalne jednačine prvog reda: **y' = z** i **z' = − y − 1/2 z**. Ako za opseg promene nezavisne promenljive **x** izaberemo (0,10) sa korakom 0.2, rešenje u obliku liste uređenih trojki **{x, y, z}** dobićete naredbom

izlaz = rksist [{z, −y −z/2}, {x,y,z}, {0,1,1}, 0.2, 10]

Dijagrame funkcionalnih zavisnosti **y(x)**, **z(x) = y'(x)** i **z(y)** dobijamo naredbama

rkplot [izlaz, 1, 2] ; rkplot [izlaz, 1, 3] ; rkplot [izlaz, 2, 3] ;

Prosto i moćno, zar ne?

DOBRO DOŠLI NA BITNET

Kako opisati jednu od najvećih računarskih mreža na svetu? Postoje bar dva popularna koncepta. Ali, kao što ćemo videti, mreži može da se priđe i na treći način.

Jedan način na koji možete opisati mrežu je da potražite najčešće korišćene operacije i procedure i kažete, recimo, da se elektronska pošta šalje na sledeći način:

MAI ^M S ^M JN% adr ^M ^M tema ^M poruka ^Z ^Z

Uz dodatno objašnjenje da **adr** označava adresu i da umesto **tema** i **poruka** treba staviti konkretan tekst, moguće je koncipirati korisno stručno uputstvo puno praktičnih kratkih komandi koje možda ne razume svako, ali koje vremenom svi nauče da koriste. Tu je i drugi koncept. On se svodi na to da dugo pričamo a da ništa zaista korisno ne kažemo. Ni jedna od navedenih alternativa ne daje dovoljno znanja o onome o čemu se priča. Ovaj tekst predstavlja pokušaj da se mreži priđe na treći način, bez mistifikacija i čudnih komandi, sa dovoljno konkretnih podataka, ali i sa pričom koja naravno iza svake mreže mora da postoji.

Šta čini jednu računarsku mrežu?

Odgovor na ovo pitanje nije jednoznačan. Puno je činilaca koji utiču na formiranje našeg viđenja jedne mreže. Neki od njih su:

- osnovne mrežne usluge
- servisi koji su raspoloživi u mreži
- fizičke veze među čvorovima mreže i performanse mreže
- tip mašina koje u mreži rade
- protokoli koje mreža koristi
- geografski prostor koji mreža pokriva
- pravila i ograničenja koja se moraju poštovati
- administracija koja održava mrežu
- koji ljudi su korisnici mreže?
- veze sa drugim svetskim mrežama
- mogućnosti priključenja u mrežu
- da li se usluge plaćaju i koliko?
- upotrebna vrednost mreže

MREŽNI PROTOKOLI

Početkom sedamdesetih godina jedna grupa IBM-ovih stručnjaka, nezadovoljnih tada postojećim rešenjima, razvila je praktično na svoju ruku (eto, i tako se radilo kod IBM-a!) novi skup protokola za komunikaciju među umreženim IBM-ovim mašinama. IBM je prihvatio ove protokole i počeo da ih distribuira pod imenom NJE (*Network Job Entry*). Da stvar bude čudnija, ali treba verovati starim spisima, softver je bio besplatan!

Network Job Entry je originalno razvijen za prenos poslova (*job files*) između IBM-ovih velikih sistema, čuvenih „System/360" i „System/370". Tipična radna okolina u kojoj je radio NJE izgledala je ovako: centralni sistem koji obavlja obradu u hladu i staklu kao što je red, povezan je sa većim brojem manje ili više udaljenih mesta na kojima se pripremaju podaci za „gladno čudovište koje jede brojeve". Za pripremu podataka i raspodelu rezultata korišćene su se instalacije poznate pod imenom RJE stanice (*Remote Job Entry Workstation*).

Svaka RJE stanica sastojala se od bušača kartica, čitača kartica, štampača i hardvera koji je celu tu veliku mašineriju kontrolisao. Tu je

Zvonko Oršolić



bio i teleprinterski terminal za kojim je sedao operater. Njegova dužnost je bila da primi bušene kartice od korisnika, da ih stavi u čitač kartica, da izda komande za prenos podataka koji se nalaze na karticama na centralni računar. Pored toga, operater je bio zadužen da kontroliše štampačke redove (ti redovi su na centralnom računaru!), da zahteva štampanje na lokalnom štampaču, ponavljanje tekuće štampe ili dela tekuće štampe (ukoliko dođe do gužvanja papira), da distribuira korisnicima odštampane listinge i na bušaču izbušene kartice.

Jasno je da je RJE stanica velika instalacija (tu je i više karitčnih bušilica, svaka veličine radnog stola) koja omogućava korisnicima da koriste veliki računar bez potrebe da do njega lično odu. Uvođenje ovakvih stanica bio je nesumnjivi napredak u istoriji računarstva, jer je povećavalo iskorišćenje enormno skupih računara. Komunikacione potrebe ovakve okoline svodile su se na potrebu da se prenesu podaci od čitača kartica do centralnog računara, od centralnog računara do bušača kartica i od centralnog računara do štampača. Pored ovoga, bilo je potrebno da se obezbedi mogućnost da operater preko iste veze može brzo da pošalje kratku kontrolnu poruku računaru. Upravo to i tako se radi u okviru NJE protokola.

Stare RJE stanice je odnela istorija. Zamenile su ih mnogo moćnije radne stanice. Kada je trebalo da se nove stanice uključe u stare sisteme, tu je bio već prihvaćeni softver (NJE) pa su nove stanice počele da emuliraju stare. Tako su protokoli i za njih vezana terminologija preživeli i stigli i do naših dana.

NJE protokoli podržavaju *peer-to peer* komunikacije (komunikacije ravnopravnih procesa). Nijedan od učesnika u NJE okolini ne traži dozvolu za slanje ni od koga. Svako svakome može slati podatke. Međutim, radi zaštite računara od neautorizovanog startovanja poslova ipak se od strane administracije računara uvode ograničenja.

NJE okolina omogućava neke veoma praktične primene. U NJE okolini možete na jednom računaru napraviti datoteku sa specifikacijom posla koji treba obaviti i tako specificirani zahtev staviti u red za izvršavanje poslova na nekoj drugoj mašini. Po obavljenom poslu mašina koja je posao obavila rezultate (izveštaje) šalje kao datoteku korisniku koji je zahtev podneo. Sve to može biti praćeno interaktivnim porukama o obavljenom poslu (ukoliko je to predviđeno), kako operaterima, tako i korisnicima na raznim mašinama. Štaviše, posao se može organizovati tako da se deo posla obavi na jednoj mašini, a da ona potom drugi deo posla prosledi na izvršavanje nekom drugom (sporijem ili bržem) računaru. Sve to zvuči dobro i u današnje vreme. Pod pojmom „računar" ovde podrazumevamo nešto što je IBM kompatibilno (u smislu IBM System/360, a ne IBM PC kompatibilno!)

I posle svih ovih godina NJE je ostao veoma popularan način za razmenu podataka, razmenu poslova i upravljanje štampačima kada se radi sa velikim IBM mašinama. Na raznim operativnim sistemima ovaj protokol izvršavaju razni softverski paketi. Na *Multiple Virtual Storage* (MVS) operativnom sistemu taj posao obavlja *Job Entry Subsystem* (JES). Na operativnom sistemu *Virtual Machine/370* (VM) to radi *Remote Spooling Communications Subsystem* (RSCS).

Zbog svoje pouzdanosti i robustnosti NJE je izašao iz sfere svoje osnovne namene. Pogotovo od pojave operativnog sistema VM (1972) NJE se intenzivno koristio i kao sredstvo za prenos datoteka sa proizvoljnim sadržajem i kao sredstvo za prenos interaktivnih poruka među interaktivnim korisnicima sistema. Dva interaktivna sistema, *Conversational Monitor System* (CMS) na VM-u i *Time Sharing Option* (TSO) na MVS-u koriste NJE za ove svrhe. Tako je NJE postao važno sredstvo za prenos datoteka bez obzira na to kakvog su oni tipa. Nije trebalo dugo čekati da postane i nosilac elektronske pošte. IBM-ov *Professional Office System* (PROFS) koristi NJE za prenos pošte, dokumenata i interaktivnih poruka.

U početku se NJE koristio na linijama na kojima je kao niži komunikacioni sloj išao BSC (*Binary Synchronous Communications*) protokol, IBM-ov protokol drugog sloja. NJE je u tim situacijama obavljao ulogu komunikacionog softvera, to jest današnjih viših slojeva OSI modela. U novije vreme NJE je definisan tako da radi povrh SNA, TCP/IP, OSI i DECnet veza. Kao takav, pre se može smatrati mrežnom aplikacijom.

PRIMI I PROSLEDI

NJE je protokol van IBM-ovih zvaničnih arhitektura. Često se označava i kao LU0 (logical unit 0). U knjigama o SNA mrežama ne mogu

se naći materijali o ovom tipu komunikacija.

Svaki uređaj (računar, RJE radna stanica...) koji učestvuje u izgradnji NJE mreže naziva se čvor (node). Svaki čvor ima svoje ime. Čvorovi su međusobno vezani fizičkim vezama tipa *point-to-point* koje ćemo zvati linkovi (*links*). Link uvek nosi ime po čvoru do koga vodi. Na primer, ako je čvor A direktno povezan sa čvorom B, čvor A će link prema računaru B zvati B, a čvor B će link prema čvoru A zvati A.

Link može biti u jednom od sledeća tri stanja:

- INACTIVE: računar koji daje izveštaj o stanju linka ne pokušava da uspostavi vezu.
- ACTIVE: računar koji daje izveštaj pokušava da uspostavi vezu, ali je ona u prekidu.
- CONNECT: postoji veza između računara koji daje izveštaj i računara koji ima isto ime kao i link.

Svaki čvor u mreži poseduje tabelu koja služi za potrebe nalaženja puta do odredišnog čvora. Tabela rutiranja (*routing table*) sadrži imena (svih!) čvorova u mreži i linkova koje treba koristiti da bi se do tih čvorova stiglo. Svaki čvor koji primi podatke konsultuje tabelu rutinga da bi utvrdio kome treba da prosledi primljene informacije. Ukoliko je poruka namenjena tom čvoru ili ukoliko do odredišnog čvora nije definisan link, poruka se isporučuje nekom od lokalnih korisnika ili servera. Nije dozvoljeno da dve mašine unutar iste NJE mreže imaju isto ime. Budući da mašina ima mnogo, problem imena rešava se uz asistenciju centralne administracije, koja odobrava ili ne odobrava ime za svaku novu mašinu.

NJE mreže obavljaju prenos podataka po principu „primi pa prosledi" (*store & forward*). Na putu ka svome odredištu podaci putuju kroz više mašina posrednika. Svaka od njih čeka da primi kompletne podatke (datoteku), pre nego što započne prenos ka sledećoj mašini u nizu.

NJE mreže nemaju dinamički ruting. Kao što je napred navedeno, unapred se (statički, na osnovu tabele rutinga) zna kojim putem će poruka proći. Postoji samo jedan put između dva računara u mreži. Ukoliko se taj put prekine, mreža ne pokušava da pronađe alternativni put, čak ni u situacijama kada postoje paralelne fizičke veze. U tim situacijama datoteke koje su u transportu stoje u redovima i čekaju!

Šta NJE nudi? U NJE mrežama možemo koristiti sledeće osnovne mrežne usluge:

- Elektronsku poštu
- Prenos datoteka u smeru od lokalnog računara ka udaljenom računaru
- Prenos interaktivnih poruka.

Ovakve mreže **ne nude**, osim na ponekim mašinama, sledeće mrežne usluge:

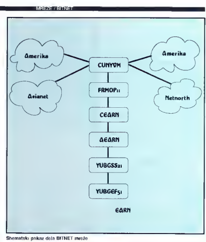
- Virtuelni terminal
- Mogućnost kopiranja datoteke sa udaljenog računara (čak ni kada osoba ima nalog na udaljenoj mašini!)

Zašto ovoliko priče o NJE-u? Na bazi NJE izgrađene su dve velike svetske mreže:

- velika IBM-ove privatna mreža pod imenom VNET
- čuvena univerzitetska i istraživačka mreža o kojoj je i ovde reč: BITNET

ŠTA JE BITNET

BITNET je nastao 5. maja 1981. (zna se tačan datum!). Tada je uspostavljena veza između dva velika američka univerziteta: City University (Njujork) i Yale University (Jel). Povezani su glavni računari ovih univerziteta CUNYVM i YALEVM. Veza je prvobitno bila namenjena za komunikaciju sistemaca iz računskih centara ovih univerziteta. Međutim, iz te veze izrastao je niko drugi do BITNET.



Shematski prikaz dela BITNET mreže

Kažu da je imenu BITNET na neki način kumovao NJE. Kao što smo spomenuli ranije, NJE je deljen besplatno, a „pošto se tu već našao" iskorišćen je za prvu BITNET vezu. Ako je verovati tim navodima, skraćenica je prvobitno označavala „Because It's There" Network (It se odnosilo na NJE). Nešto kasnije, pošto je ime BITNET već usvojeno, neko je promenio značenje skraćenice. Sada se tvrdi da naziv BITNET potiče od konstrukcije „Because It's Time Network".

Tokom vremena mreža se razvila. Danas u njoj ima između 3500 i 4000 računara raznih tipova. Najčešće se sreću kombinacije IBM VM/RSCS, DEC VAX/VMS/JNET, IBM MVS/JES2/JES3 i UNIX/UREP.

Glavni informacioni centar mreže je BITNIC (*BITNET Network Information Center*), lociran u Prinstonu, SAD. Ovaj centar pruža informacije članovima i potencijalnim članovima mreže, ugovara kupovinu hardvera i softvera sa značajnim popustima za članice i tome slično. Centar se izdržava od članarine.

Mreža BITNET pokriva praktično ceo svet. Geografski, u njoj se izdvajaju četiri mreže, od kojih svaka ima svoje ime:

- BITNET (u užem smislu) koji pokriva Sjedinjene američke države, Srednju i Južnu Ameriku
- NETNORTH je mreža istog tipa u Kanadi
- ASIANET pokriva Daleki istok: Japan, Singapur, Tajvan, Južnu Koreju
- EARN (European Academic and Resarch Network) u koji spadaju Evropa, Afrika i Bliski istok

PRIRODA MREŽE

Sve ove mreže karakteriše ista tehnička osnova (NJE). Administrativno, radi se o više mreža. Funkcionalno, to je jedna mreža. S obzirom da običnog korisnika više interesuje funkcionalna nego administrativna strana mreže, mi ćemo u daljem tekstu sve ove mreže označavati jednim jedinim imenom: BITNET.

BITNET je kooperativna mreža. To je mreža u koju dobrovoljno ulaze organizacije članice. To nije profesionalna mreža za prenos podataka. Ona se više zasniva na dobroj volji članica da budu učesnici jednog velikog projekta koji svima pruža dobre komunikacione usluge. Postoji skup fer-plej pravila koja bi trebalo poštovati, ali i malo garancije da će sva biti sprovedena. Tako koncipirana mreža, izvesno je, ne može da garantuje sigurnu isporuku ili tajnost podataka. Stvar je profesionalne časti i realnih tehničkih uslova koje srećemo na različitim lokacijama, da se što bliže priđe visokim standardima tog tipa. Međutim, obaveze i obećanja niko ne daje.

Poput ostalih univerzitetsko-istraživačkih mreža, i BITNET spada u mreže koje služe za primenu, testiranje i demonstriranje uspešnosti novih tehnologija i proizvoda iz oblasti komunikacija i računarstva. Tehnološki, mreža se uporno, iz godine u godinu, značajno menja.

Novi tipovi i brzine veza i novi tipovi komunikacionih računara deo su svakodnevice BITNET-a.

Interesantnije članice BITNET-a su veliki univerziteti i univerzitetski računski centri, državne istraživačke institucije i neke velike kompanije primljene u BITNET po posebnim kriterijumima.

FIZIČKE VEZE

Računari unutar BITNET-a vezani su međusobno na više načina. Najčešće se kao veza koriste iznajmljene telefonske linije, ali se sreću i komutirane telefonske veze, satelitske veze, digitalne veze, veze preko lokalnih mreža i drugo. Brzina komunikacija na tim vezama varira od veze do veze i kreće se od 1200 b/s do 2Mb/s.

U poslednje dve godine naročito intenzivno se radi na prestrukturiranju mreže. Po rečima *Mirjane Tasić*, trenutno glavni trend u EARN-u su pravljenja višestruko povezane glavne magistrale regionalnih računara (EARN backbone) i uvođenje elemenata dinamičkog rutinga u mrežu, čime bi se eliminisala neka od najvećih ograničenja EARN-a.

STANJE U SRBIJI

Pošto je Srbija u Evropi, mi spadamo u deo BITNET-a u koji se zove EARN. Sredinom 1988. godine proradio je nas prvi računar u BITNET-u. Bio je to YUBGSS21, Nacionalni čvor za Jugoslaviju, fizički smešten u Republičkom zavodu za statistiku Srbije (RZSS). Radi se o skupoj i dobro održavanoj mašini, o velikom IBM ovom sistemu (trenutno je to IBM 3090), koja je ušla „u igru" zahvaljujući tesnoj saradnji u nabavci i korišćenju opreme između RZSS i Prirodno-matematičkog fakulteta iz Beograda.

Početkom 1990. proradio je i drugi BITNET računar. Bio je to YUBGEF51, MicroVAX 3600, smešten na Elektrotehničkom fakultetu u Beogradu. Radi se o mašini koja je priključena na dve mreže: na DECnet, koji se koristi od strane projekta „Sistem naučno tehničkih informacija Jugoslavije" (SNTIJ), i na BITNET. Mašina je zamišljena kao mrežni prolaz između ove dve mreže (spaja DECnet sa BITNETom). Unutar DECnet mreže ista mašina poznata je pod imenom UBBG (skraćeno od Univerzitetska biblioteka „Svetozar Marković" Beograd). UBBG obavlja dva važne uloge: to je BITNET *gateway*, a istovremeno i mašina koja se koristi za potrebe automatske obrade podataka u gore navedenoj i još nekim bibliotekama!

EVROPSKA MREŽA

EARN (European Research and Academic Network) je asocijacija registrovana u Francuskoj, specijalizovana za organizovanje i održavanje istoimene računarske mreže u Evropi, Africi i na Bliskom istoku. Članice ove organizacije su države. Asocijacijom upravlja Savet direktora (*Board of Directors*), organizovan tako da iz svake zemlje članice ima po jedan predstavnik. Nas predstavlja profesor *Jagoš Purić* sa Prirodno-matematičkog fakulteta, inače direktor EARN Jugoslavija. Savet direktora bira Predsednika EARNa i Izvršni komitet (*Executive Committe*), koji dalje biraju Tehnički komitet EARN-a i Komitet za prijem novih članova (*Membership Commity*). Mrežom se administrativno upravlja iz Pariza.

Centralni tehnički i administrativni servis organizovan je za svaku zemlju ponaosob, i to od strane administracije centralnog računara za tu zemlju. Zvanično ime tog računara je „Nacionalni čvor". Najodgovornija osoba za Nacionalni čvor zove se Nacionalni koordinator (*National Country Coordinator*). Njena uloga je da, pored ostalog, posluži kao veza između centralnih delova EARNa i administratora čvorova mreže u svakoj od država (i to u oba smera). Ova osoba je takođe centralni izvor informacija o EARN-u i tehnički veoma engazovana oko Nacionalnog čvora. Nacionalni koordinator za Jugoslaviju je gospođa *Mirjana Tasić* iz Republičkog zavoda za statistiku Srbije.

Svaka zemlja plaća liniju do najbliže zemlje koja već ima vezu sa EARNom. Veze se ostvaruju tako što se priključuje Nacionalni čvor na (obično) Nacionalni čvor, a od Nacionalnog čvora za svaku zemlju veze se dalje zvezdasto granaju.

Mreža se izdržava od članarine. Članarinom se finansiraju veze sa Amerikom i centralna administracija EARN-a. Članarinu plaćaju zemlje članice, i to proporcionalno nacionalnom dohotku. Unutar svake zemlje rešava se, zavisno od ekonomske situacije i društvenog poretka, način plaćanja članarine. Negde plaća država, a negde organizacije članice. Individualni korisnici, uobičajeno je, ne plaćaju usluge EARN-a, ili da budemo konkretniji, ne učestvuju u plaćanju članarina za EARN.

Asocijacija EARN osnovana je 1983. po uzoru na američki deo BITNETa. Do 1987. godine IBM je finansirao veze sa Amerikom. Kasnije se donacijama istakao „Digital Equipment Corporation" (DEC).

U novije vreme EARN migrira ka novim standardima, preciznije uvode se NJE po X.25 vezama i NJE po TCP/IP vezama. U ovim naporima ostvaruje se veoma bliska saradnja sa svetskim računarskim i telekomunikacionim džinovima kao što su DEC, IBM i Northern Telecom. Migraciju sprovodi EARN u koordinaciji sa Evropskom asocijacijom računarskih mreža RARE. Cilj je da velike evropske mreže – EARN, EUnet i HEPnet – iz razumljivih ekonomskih i tehnoloških razloga dele istu komunikacionu infrastrukturu.

Na slici je prikazano stanje veza u mreži koje smo zatekli u trenutku priključenja UBBG-a u BITNET.

IMENA ČVOROVA I KORISNIKA

Kao što smo pomenuli, imena svih čvorova u BITNET-u su jednoznačna. Pri davanju imena obično se poštuju neka pravila koja se razlikuju od kontinenta do kontinenta. U Evropi važi da Nacionalni čvor nosi ime koje se dobija „lepljenjem" niza EARN na ISO oznaku države kojoj pripada. Po toj logici čvor za Jugoslaviju trebalo je da se zove YUEARN. Računar u RZSS nema mogućnosti da obavlja sve poslove Nacinalnog čvora. On je zamišljen kao dobro privremeno rešenje, pa nije uzeo ime generisano na taj način. Deo poslova Nacionalnog čvora za nas još uvek obavlja austrijski nacionalni čvor AEARN u Lincu. Očekuje se da će tokom 1992. poslove nacionalnog čvora preuzeti novi univerzitetski računar IBM 3090, koji će biti snabdeven svim potrebnim softverom da bi takvu ulogu obavio.

Naši računari dobili su imena po „nemačkim" pravilima koja su unutar EARN-a formalizovana u cilju izbegavanja duplih imena. Prva dva slova označavaju zemlju, druga dva grad a treća dva instituciju u kojoj se čvor nalazi. Zatim sledi jedna cifra koja označava tip mašine i operativnog sistema (1 za IBM/VM, 2 za IBM/MVS, 5 za VAX/VMS). Poslednja cifra služi za razlikovanje u slučaju da imamo više mašine istog tipa na istoj lokaciji. Tada prva mašina na toj poziciji ima broj 1, druga broj 2 i tako redom.

Imena računara u BITNETu imaju najviše 8 znakova. Nekoliko primera imena čvorova imamo na slici. Tu su AEARN (Austrijski nacionalni čvor), CEARN (računar u CERN-u, Švajcarska) i FRMOP11 (Računar u Monpeljeu, Francuska) od koga ide veza sa Amerikom.

Korisnička imena takođe imaju do 8 znakova. Ukoliko neka od mašina dozvoljava duža imena korisničkih naloga (*Username*), samo prvih osam znakova su značajni. To može da stvori ozbiljne probleme ukoliko postoje duplikati korisničkih imena na skupu od prvih osam znakova.

Adresa korisnika unutar mreže piše se na sledeći način:

username@node

gde je **node** ime čvora, **username** ime naloga (**login id** ili kako se to već sve zove), a @ razdvajač ova dva elementa adrese. Ukoliko zanemarimo problem predugih **username**-ova (malopre spomenuti), svaki korisnik unutar mreže ime jednoznačnu adresu.

AKO PADNE JEDAN...

Pogledajmo još jednom sliku i specijalni položaj računara CUNYVM na njoj. Ovaj računar se nalazi u centru BITNET-a. To je tačka u kojoj se susreću više američkih delova BITNETa (delova koji nemaju drugih međusobnih veza!), veze prema Dalekom istoku, veze prema Evropi i veze prema Kanadi! Pad ovog računara, zahvaljujući organizaciji BITNET-a, izazivao je raspad mreže na više delova koji i dalje rade, ali više nemaju međusobnih veza. Zahvaljujući mehanizmu „primi pa prosledi" poruke upućene putanjama kroz mašinu CUNYVM nisu se gubile, ali nisu mogle ni da prođu do odredišta. Zamislite tek tragične posledice koje bi izazvao duži kvar ovog računara. To se, srećom, retko dešavalo (CUNYVM je jedna od najvećih računarskih instalacija u svetu).

Datoteka se, pre nego što se proslede do sledećeg čvora posrednika, skladište na svakom čvoru duž puta. Za razliku od poruka, koje se u slučaju prekida veze gube, datoteke se ne gube. Na svakom čvoru postoji red čekanja (**queue**) u kome se beleže podaci o datotekama koje su pristigle, još nisu prenele dalje i čekaju na dalji prenos. Po izvršenom prenosu datoteka i informacija o tome da je treba preneti brišu se sa računara koji su poslužili kao posrednici za prenos datoteke.

Ono što se retko dešava sa CUNYVM dešava se mnogo češće i sa ostalim računarima unutar BITNET-a. Mašine, kao što im i priliči, padaju, rade bekape, servisiraju se, kvare se. Svaki prekid rada mašina koja služi kao prolaz porukama sa drugih mašina odseca od sveta deo mreže koji od te mašina zavisi. Pored ovoga srećemo i niz drugih problema. Najvažniji su preopterećenja komunikacionih linija i preopterećenja računara. Uska grla koja se na taj način stvaraju izazivaju zastoj kompletnog BITNET saobraćaja.

Uzgred, današnja konfiguracija veza do Amerike nešto je drugačija nego što je to na slici prikazano. Trenutno je austrijski nacionalni čvor AEARN direktno povezan sa Univerzitetom u Džordžiji (čvor UGA), pa sva pošta za Ameriku ide tim putem.

Novi projekti BITNET II i EARN II izvršili su geografsko grupisanje čvorova oko glavnih magistrala, pri čemu su članovi magistrale vezani višestrukim vezama. Magistralni čvorovi izvršavaju i nešto drugačije algoritme za ruti-

nja saobraćaja. Time se pouzdanost prenosa povećala, a opterećenje mreže ravnomernije rasporedilo.

Elektronska pošta između Beograda i Amerike prenosi se duž BITNET a za vreme koje varira između jednog minuta i više dana! Najuobičajenija vremena su reda veličine nekoliko minuta. S obzirom na napredak u organizaciji mreže, mogu se očekivati i bolji rezultati.

DATOTEKE I PORUKE

NJE prenosi datoteke (NJE files) i poruke (zvanično *Nodal Message Records* ili NMR). Mreža tretira sve poslate podatke ili kao datoteke ili kao poruke.

Poruke su pojedinačni redovi teksta koje se šalju na lokalni ili udaljeni računar u mreži. Dužina jedne poruke je do 160 bajtova. Poruke adresirane na konkretnog korisnika ili server nazivamo interaktivnim porukama (*interactive messages*). Poruka se može poslati i bez specificiranja korisnika kome je upućena (blanko korisnik). Takva poruka tretira se kao komanda mrežnom softveru mašine na koju je upućena i zove se mrežna komanda (*network command*).

Treći tip poruka je odgovor na komandu (*command reply*). Ovakve poruke šalju mrežni serveri kao odgovor na mrežne komande. Mreža na jednak način tretira sve poruke, bez obzira na njihovo poreklo ili sadržaj. Poruke se prosleđuju i isporučuju korisnicima odmah po prijemu. One se ne skladište u mreži. Prenose se istovremeno sa datotekama (imaju prioritet u odnosu na datoteke) tokom prenosa datoteka, tako da kroz mrežu putuju veoma brzo.

Poruke upućene interaktivnim korisnicima ispisuju se na ekranu korisnika ukoliko on trenutno radi. Ukoliko korisnik kome je poruka upućena trenutno ne radi, računar koji je poruku primio sastavlja novu interaktivnu poruku kojom javlja pošiljaocu da traženi korisnik nije trenutno prisutan.

Mrežne komande (poruke za blanko korisnika) su najčešće zahtevi za dobijanje informacija tipa broja prijavljenih korisnika, tačnog vremena, statusa komunikacionih linija i slično. Većina sistema u BITNET u odgovara na njih.

Sa VAX/VMS/JNET sistema, kakav je i UBBG, poruke se šalju sledećom komandom:

$ SEND username@node poruka

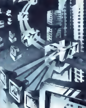
Moguće je napisati programe koji primljene poruke „hvataju" i na njih reaguju na predviđeni način (recimo šalju odgovore). Programi ovakvog tipa zovu se mrežni serveri (*network servers*) ili prosto serveri.

MREŽNE KOMANDE

Mrežne komande nisu standardizovane. Mogu se razlikovati (i razlikuju se) od mašine do mašine. Neke mašine su voljne da odgovaraju na komande koje im šaljete (poznate kao „pričljive" mašine). Neke druge su mnogo manje raspoložene za priču.

U BITNET-u ubedljivu većinu čvorova (oko 75%) čine sistemi IBM/VM/RSCS i DEC/VMS/JNET. I jedni i drugi spadaju u mašine spremne za razgovor, tako da se sa njima možete siti napričati. I jedni i drugi reaguju na mrežne komande koje su navedene u nastavku.

Kao što je ranije navedeno, komande se šalju tako što se u komandi SEND navede mrežna komanda koju šaljemo. Recimo, ukoliko sa UBBG-a želite da pitate čvor YUBGEF51 (da, to je ista mašina) koji mrežni softver radi na njemu, poslaćete mu mrežnu komandu



QUERY SYSTEM LEVEL i to na sledeći način:

$ SEND @YUBGEF51 QUERY SYSTEM LEVEL

Ukoliko isto pitanje želite da postavite računaru AEARN, uradićete to komandom

$ SEND @AEARN QUERY SYSTEM LEVEL

YUBGEF51 je tipičan VAX/VMS sistem, AEARN je tipičan VM sistem. Eto lepe prilike za vežbanje. Praktično svaki čvor kome znate ime može vam po nešto odgovoriti.

Značajnije komande koje primaju i VAX/VMS i VM sistemi su:

QUERY SYSTEM LEVEL – daje verziju komunikacionog softvera i operativnog sistema koji na mašini radi.
QUERY SYSTEM LINKS – prikazuje sve linkove za koje postoje definicije na mašini, uključujući i neaktivne (INACTIVE) linkove.
QUERY SYSTEM ACTIVE – prikazuje sve aktivne (CONNECT i ACTIVE) linkove.
QUERY node – kom čvoru će ovaj čvor poslati podatke ukoliko dobije podatke adresirane na čvor koji ste mu naveli umesto node.
QUERY link FILES – koliko datoteka čeka u redu vezanom za link link.
QUERY link QUEUE – daje informaciju koliko datoteka čeka u redu vezanom za link link i njihov spisak! (Pazite, komanda opterećuje mrežu!)
CPQUERY USERS – daje broj trenutno prijavljenih korisnika na mašini.
CPQUERY NAMES – daje spisak korisnika koji trenutno na mašini rade.

Sledeće mrežne komande su specifične za VAX ove sa VAX/VMS om i JNET om. To jest za YUBGEF51 (UBBG).

FINGER – spisak trenutno prisutnih korisnika, imena programa koje trenutno izvršavaju, koliko vremena radi i na kom terminalu. (I sve to baz ikakvih privilegija!)
SHOW USERS – daje potpuno isti izveštaj.

Specifična za YUBGSS21 (koji spada u mašine koje ne vole mnogo da razgovaraju) je sledeća komanda:

$DA,T – daje spisak trenutno aktivnih korisnika.

IGRE BEZ GRANICA

I za kraj, pogledajmo nekoliko praktičnih primera korišćenja raznih varijeteta komanda SEND. Komande koje su ovde date primenjene su na računaru UBBG. Odgovori su oplemenjeni editorom (izdvojene su važnije informacije, a višak uklonjen).

Interaktivna poruka za Nacionalnog koordinatora.

$ send system3@yubgss21 Dobar dan Miro

Interaktivna poruka za autora ovog teksta.

$ send zvonko@yubgef51 je si li čitao poruke na sezamu?

Pitanje računaru YUBGEF51: Koji mrežni softver koristiš?

$ send @yubgef51 query system level
(YUBGEF51) – Jnet V3 4 router V4 34

Za YUBGEF51: Da li ti je veza sa RZSS u stanju Connect?

$ send @yubgef51 query system active
(YUBGEF51) – Link YUBGSS21 Connect
(YUBGEF51) – Link YUZGRB51 Active

Za YUBGEF51: Kome ćeš proslediti poruku upućenu na mrežni informativni centar (čvor BITNIC)?

$ send @yubgef51 query bitnic
(YUBGEF51) – BITNIC routed through link YUBGSS21

Za YUBGEF51: Koliko datoteka čeka na isporuku prema Nacionalnom YU čvoru?

$ send @yubgef51 query yubgss21 files
(YUBGEF51) – Link YUBGSS21 Queued: 1

Za YUBGSS21: Ko sve trenutno radi? (Ili jesi li živ?)

$ send @yubgss21 $da t
(YUBGSS21) – 15 27 48 TSU01077
$HASP608 JAT0025
(YUBGSS21) – 15 27 48 TSU01077
$HASP608 JAT0025
(YUBGSS21) – 15 27 48 TSU01826
$HASP608 H00023D
(YUBGSS21) – 15 27 48 TSU01826
$HASP608 H00023D

Za AEARN: U kakvom su stanju tvoje komunikacione veze?

$ send @aearn query system links
(AEARN) – LINK CEARN CONNECT
(AEARN) – LINK DEARN CONNECT
(AEARN) – LINK HEARN CONNECT
(AEARN) – LINK FRMOP11 CONNECT
(AEARN) – LINK UGA CONNECT
(AEARN) – LINK CSEARN CONNECT
(AEARN) – LINK HUEARN CONNECT
(AEARN) – LINK YUBGSS21 CONNECT

Za AEARN: Koma ćeš poslati Mail koji ja šaljem na BITNIC?

$ send @aearn query bitnic
(AEARN) – BITNIC ROUTED THROUGH LINK UGA

Za AEARN: Kolika je gužva na putu ka čvoru UGA? Koliko datoteka čeka na isporuku?

$ send @aearn query uga files
(AEARN) – LINK UGA CONNECT
SENDING=7 RECEIVING=0 QUEUED=411

Lepo za igru, zar ne?

Bio je ovo uvod u BITNET. Upoznali smo se sa nekim od principa na kojima je mreža zasnovana, sa njenom prošlošću i sadašnjošću. Još uvek nismo govorili ništa o najvažnijim uslugama koje se putem BITNET a mogu dobiti: o elektronskoj pošti, prenosu datoteka, o serverima i mrežnim prolazima. BITNET je priča bez kraja, pa se ne treba truditi da cela stane u nekoliko strana. Očekujte nastavak. Može?

$ send @aearn query system links
(AEARN) – LINK CEARN CONNECT

REČI SVUDA OKO NAS

Predlažem vam da pročitate ovaj tekst, iako ste možda vatreni pristalica računarstva kao čisto praktične discipline. U njemu će biti reči o formalnim jezicima, matematičkim automatima i sličnim napravama iz galerije teorijskog računarstva. Namera nam je da objasnimo osnovne pojmove iz teorije formalnih jezika, teorijske discipline koja ima ogromnu primenu, i to ne samo u računarstvu.

Čak i ako se zanemari veliki matematički značaj ove discipline, verovatno će vas interesovati njena izuzetna primenljivost u praksi. Si gurno ste već čuli pojmove poput formalna gramatika, leksička analiza i slično. Ovaj tekst vam može pomoći da lakše razumete osnovne pojmove iz ove oblasti, što će vam pre ili kasnije zatrebati. Ako pravite svoj prevodilac ili neki sličan program, takva literatura vas ne može mimoići. U krajnjem slučaju, ovaj tekst možete shvatiti i kao uvod u jednu veoma interesantnu igru.

REČI I JEZICI

Naša igra počinje sasvim neobavezno: pravićemo nove reči i nove jezike. Ovakav zadatak zvuči dosta neobično: napraviti jedan pravi jezik sa svim elementima nikako nije posao za jednog čoveka. Prirodni jezici uključuju u sebi mnoge veoma složene pojave kao što su istorija, značenje itd., koje nas ovde ne interesuju. Ovde se govori o **formalnim jezicima**, koji su u krajnjoj liniji jednostavni skupovi stringova (reči). Ako se pitate otkud termin **jezik**, osvrnite se oko sebe: praktično ceo svet funkcioniše kao skup objekata koji na neki način komuniciraju. A za opis njihove komunikacije teško ćete naći prikladniji izraz od **jezik**.

Jezik se može posmatrati i kao skup nekih reči. Postoji skup simbola (po analogiji se taj skup može zvati **alfabet**) čijim se nizanjem dobijaju **reči**. To znači da je sledeći skup jedan jezik:

a, ab, ba, aba

Ovaj jednostavan jezik se sastoji samo od četiri reči i sve reči su sastavljene od simbola **a** i **b** koji čine alfabet. Naravno, jezici koje mi definišemo ne moraju biti ovako jednostavni – u većini slučajeva će to biti daleko komplikovaniji skupovi. U svakom slučaju, osnovni principi ostaju isti – u ovako jednostavan koncept se uklapa velika grupa jezika koji se koriste u praksi.

Kada definišete svoj jezik, morate ga na neki način opisati – jednostavno nabrajanje njegovih reči obično nije efikasno. Pogledajmo neke efikasnije načine.

REGULARNI JEZICI

Kada vam zatreba da izbrišete sve nepotrebne datoteke iz tekućeg kataloga, otkucaćete bez mnogo razmišljanja nešto poput **del *.bak**. Ovim ste na krajnje jednostavan način iz skupa svih datoteka u katalogu izdvojili jedan **podskup** na koji treba primeniti operaciju brisanja. Koristeći prethodnu terminologiju, izrazom ***.bak** vi ste opisali jedan jezik.

I početnici veoma brzo usvajaju razne džoker simbole koji omogućavaju jednostavno specificiranje grupe datoteka. Komanda **del a??*** veoma pregledno opisuje brisanje svih datoteka čiji naziv ima tri slova i počinje slovom **A**. Korisnici nekih proširenja MS DOS-a mogu koristiti finiji mehanizam za određivanje grupe datoteka. Na primer, ako otkucate 4DOS komandu

"Bezbojne zelene ideje spavaju besno" – Noam Čomski, primer iz knjige „Sintaksičke strukture"

Nenad Batočanin

del [pft]*

izbrisaćete sve datoteke iz tekućeg kataloga čiji naziv počinje slovima **p**, **f** ili **t**. Da bi se izbeglo nabrajanje veće grupe simbola, može se pisati i nešto poput

del [a-h]*.tmp

Ova komanda briše sve datoteke čiji naziv počinje slovima **a**, **b**, **c**, ..., **h** a imaju ekstenziju **TMP**. Zapis **[a-h]** označava sve simbole između **a** i **h** po redosledu u **ASCII** tabeli. Zapis **[!a-z]** označava bilo koji simbol koji **ne** pripada skupu od **a-z**. Tako se sve datoteke čiji naziv ne počinje cifrom dobijaju sa

dir [!0-9]*

Korisnici kvalitetnijih editora imaju sličnu mogućnost za pretraživanje i zamenu teksta: simboli * i ? samo su deo mnogo bogatijeg arsenala. Na primer, u editoru *BRIEF* se regularni izrazi koriste za zadavanje teksta koji treba tražiti u datoteci. Sledećim izrazom zadaju se sve naredbe **if** i **while** koje počinju proizvoljnim brojem simbola *tab*:

<\t@{**if**}|{**while**}

Simbol < označava početak reda, **\t** simbol *tab*, @ proizvoljno ponavljenje prethodnog simbola, a {**if**}|{**while**} da posle toga dolazi **if** ili **while**. Slične konstrukcije se koriste i u programima za pretraživanje datoteka (**Grep**) i u nekim specijalizovanim programskim jezicima (na primer **AWK**).

U osnovi, svi opisani sistemi imaju isti, naizgled jednostavan zadatak: zadavanje ili opis nekog skupa stringova, koga možemo smatrati nekim jezikom. Tako komanda **del a??*** opisuje stringove **a11.dat**, **abc.c** itd. Postoji više načina za tako nešto, ali se najčešće sreće ovaj koji je upravo opisan: skup stringova (jezik) se opisuje **regularnim izrazom**. Regularni izraz je u stvari način za opisivanje jednog jezika. Jezici koji se mogu opisati regularnim izrazima nazivaju se **regularni jezici**.

KONAČNI AUTOMATI

Regularnim izrazom se *opisuje* skup stringova (odnosno jedan jezik). Korišćenjem tog zapisa možemo, na primer, saopštiti komandnom interpreteru nad kojim datotekama treba da izvrši zadatu operaciju, editoru koje reči treba tražiti u tekstu itd. Međutim, program koji izvršava zadatu instrukciju mora na neki način taj „opis" da iskoristi. Tako, ako je zadata komanda **del *.bak**, MS DOS komandni interpreter mora znati da datoteke **proba.bak** i **test.bak** odgovaraju „opisu" ***.bak**. U slučaju jednostavnih regularnih izraza stvar se može rešiti jednostavnom analizom, bez nekog naročitog komplikovanja. Ali u slučaju složenijih izraza, stvar može biti značajno teža. Na primer,

Slika 1



Grafički prikaz konačnog automata

sledeći regularni izraz je potreban da se programu za generisanje leksičkih analizatora *Lex* objasni šta je to decimalni broj sa opcionim eksponentom:

[0-9]+(\.[0-9]+)?(E[+\-]?[0-9]+)?

Zbog toga se za opisivanje jednog jezika koriste i druge (ekvivalentne) metode, a koje mogu pomoći pri praktičnoj realizaciji. **Konačni automati** se koriste kao takva alternativa. Matematička mašina (automat) razlikuje se od nečega što smo navikli da zamišljamo kad se pomene reč „mašina". Ovde se radi o apstraktnim mašinama (tzv. matematički modeli) koje se opisuju matematičkim formulama.

Konačni automati će nam služiti kao sredstvo za prepoznavanje reči koje pripadaju jeziku: na osnovu ulazne reči oni daju neki izlaz – u našem slučaju odgovor da li je reč iz jezika ili nije. Na taj način oni izdvajaju iz skupa svih zadatih reči samo ona koje pripadaju jeziku.

Stroga definicija konačnog automata deluje previše neinteresantno, sve dok se ne nacrta prikaz u obliku grafa (slika 1). Prikazani konačni automat služi za prepoznavanje sintaksno ispravnih brojeva sa opcionim eksponentom, znači reči **156**, **52.87**, **9.5e20**, ... On „radi" na sledeći način: polazi se od čvora grafa označenog sa „start" (broj 1). Zatim se redom čitaju simboli ulazne reči i pri tom se prelazi iz jednog čvora u drugi (čvorovi odgovaraju tzv. **stanjima automata**), ali samo ako postoji grana grafa sa oznakom tekućeg simbola. U suprotnom, automat staje i ne prihvata reč. Kada se dođe do čvora 2, 3 ili 6 (posebno označenih, tzv. **završnih čvorova**) i ako nema više slova u ulaznoj reči, automat staje i kaže da je prihvatio reč. Ako se u bilo kom čvoru desi da nisu pročitani svi simboli sa ulaza i da ne postoji put sa oznakom upravo učitanog simbola, automat staje i reč nije prihvaćena. Znači, uslov za prihvatanje je da su pročitani svi simboli sa ulaza i da se automat nalazi u završnom čvoru. Na primer, automat sa slike prihvata reči **1.23**, **654.0**, **123**, **123**, **123.45e–65**, a ne prihvata reči **e-12**, **1.**, **2** ili **.87**.

Ovim smo dobili još jedan način za zadavanje jezika: umesto regularnog izraza možemo napraviti ekvivalentan automat i reći da naš jezik čine sve reči koje prihvata automat. Ovaj

Slika 2

```
#include <stdio.h>

void main ()

{
   int c;
   int s = 1;

   while ((c = toupper (getchar())) != '\n' && s ) {
      switch (s) {

         // Stanja 1
         case 1
            if (isdigit (c)) s = 2;
            else s = 0;
            break;

         // Stanja 2
         case 2
            if (c == ' ') s = 3;
            else if (c == 'E') s = 4;
            else if (!isdigit (c)) s = 0;
            break;

         // Stanja 3
         case 3
            if (c == 'E') s = 4;
            else if (!isdigit (c)) s = 0;
            break;

         // Stanja 4
         case 4
            if (c == '+' || c == '-') s = 5;
            else if (isdigit (c)) s = 6;
            else s = 0;
            break;

         // Stanja 5
         case 5 :
            if (isdigit (c)) s = 6;
            else s = 0;
            break;

         // Stanja 6
         case 6
            if (!isdigit (c)) s = 0;

      }
   }

   // Da li je automat u zavrsnom stanju?
   if (s == 2 || s == 3 || s == 6)

      puts ("\nAutomat je prihvatio rec");
   else
      puts ("\nRec nije prihvacena");
}
```

Program za simulaciju rada konačnog automata

nikan ima određenu prednost mnogo je jednostavnije na osnovu zadatog automata proveriti da li data reč pripada jeziku nego na osnovu regularnog izraza. Opet, na osnovu automata je dosta teže reći koje sve reči pripadaju jeziku. U suštini, obično se regularni izraz koristi za opis jezika, a automat za prepoznavanje reči.

Konačni automati i regularni izrazi imaju istu „opisnu" moć: za sve što se može zadati regularnim izrazom postoji odgovarajući konačni automat i obrnuto. U stvari, mnogi programi realizuju traženje po zadatom regularnom izrazu tako što na osnovu regularnog izraza kreiraju ekvivalentan konačni automat koji zatim „raspoznaje" specificirane konstrukcije.

Za konstrukciju konačnog automata iz regularnog izraza postoji više algoritama. Napisati program koji po zadatom grafu simulira konačni automat nije problem – program se može napisati lako što će se direktno „pratiti" graf korišćenjem naredbi za grananje (čime se dobija vrlo nastrukturiran ali potencijalno najkraći program). Može se napraviti program koji na osnovu zadatog opisa automata simulira njegov rad, kao na slici 2. U računaru se konačni automat može predstaviti i tabelom, kao na slici 3.

Regularni jezici se koriste u tzv. leksičke analize u prevodiocima. O tome je već bilo reči u „Računarima", pa ćemo se samo osvrnuti na najvažnije stvari. Zbog lakše analize prevodi lac vrši prvo tzv. **tokenizaciju** programa koji se prevodi. To znači da određene nizove simbola sa ulaza zamenjuje jednim simbolom (tokenom) sa kojim se kasnije manipuliše. Na primer, za dalju analizu nije važno da li se u programu pojavio broj 112 421 ili 12e34 – dovoljno je svaki string sa ulaza koji predstavlja broj zameniti posebnom oznakom <**NUMBER**> što će biti sasvim dovoljna informacija za dalji rad. Program (procedura) koji vrši tokenizaciju ulaznog teksta naziva se **leksički analizator**. On se može konstruisati na više načina, ali se

Slika 3

		ULAZNI SIMBOLI				
		0-9		E	+	
STANJA	s1	s2				
	s2	s2	s3	s4		
	s3	s3		s4		
	s4	s6			s5	s5
	s5	s6				
	s6	s6				

Predstavljanje konačnog automata pomoću tabele

obično koristi konačni automati. Naime, svaki token na izlazu zamenjuje grupu reči (stringova) sa ulaza. Tu ulaznu grupu stringova možemo opisati regularnim izrazima. Na primer, izrazom

[0-9]+(\.[0-9])?

opisujemo sve decimalne brojeve u ulaznom programu. Oznaka \. je decimalna tačka, a znak pitanja označava da se deo iza decimalne tačke ne mora pojaviti. Znači, kada se na ulazu bude pojavio bilo koji broj, leksički analizator će vratiti token <**NUMBER**>. Slično se mogu napraviti regularni izrazi za sve ulazne **lekseme**: rezervisane reči, identifikatora i slično. Leksički analizator sad čita ulazni program i proverava da li neki od navedenih regularnih izraza odgovara ulazu. To je najlakše implementirati tako što se na osnovu svakog regularnog izraza konstruiše konačni automat koji prepoznaje odgovarajući jezik, pa se svi ti automati zajedno kombinuju u jedan koji prepoznaje sve ulazne reči. Ovaj posao je relativno jednostavan, ali je dosta zamoran, pa su zato napravljeni programi koji to rade umesto vas.

Jedan od najpoznatijih takvih programa je *Lex*. *Lex* na osnovu zadatih regularnih izraza (koji opisuju šta sve može doći sa ulaza automata) generiše konačni automat koji prepoznaje zadate reči. Originalni *Lex* generiše C program, ali u principu to može biti bilo koji jezik. Na slici 4 prikazan je program – specifikacija za *Lex* kojom se zadaje jedan minijaturni programski jezik koji se sastoji svega od nekoliko naredbi. Korišćena verzija *Lex*-a prilagođena je za *Turbo Pascal*.

Regularnim izrazima se opisuju reči koje leksički analizator treba da prepozna sa ulaza. Na primer, konstrukcija

id letter (**letter** | **digit**)*

deklariše sve identifikatore kao niz slova i cifara koji počinje slovom. Na osnovu ove specifikacije program *Lex* generiše *Pascal* program koji ulazni niz simbola pretvara u niz tokena. Na primer, ako je na ulazu niz

if x > 123 then begin end

generisani leksički analizator će vratiti niz tokena

<**IF**><**VAR**><**REL**><**NUMBER**><**THEN**><**LEFT**><**RIGHT**>

čime se dalja analiza daleko pojednostavljuje.

SLOBODNI JEZICI

Konačni automati i regularni izrazi nisu svemoćni u opisivanju – neki jednostavni jezici ne mogu opisati na taj način. Na primer, jezik svih binarnih brojeva oblika **111 111000 000** (broj jedinica jednak je broju nula, ovo se često piše kao 1ⁿ0ⁿ) ne može se nikako opisati regularnim izrazom niti konačnim automatom – jednostavno takav automat/izraz ne postoji.

Zato se za zadavanje jezika osim regularnih izraza i automata koriste i druga sredstva. Jedan od tih metoda zauzima posebno mesto – tzv. **gramatike**. Kao gramatika u prirodnom jeziku, tako i formalna gramatika služi za zada

Slika 4

```
%{
   { Uključivanje datoteke sa definicijama tokena }
   {$I Tokens}
%}

/* Definicije osnovnih leksema */

space    [ \t]+
letter   [A Za z]
digit    [0-9]
id       {letter}({letter}|{digit})+
num      {digit}+
rel      [<>=]

%%

/* Akcije koje treba urediti kad se naidje */
/* na odgovarajuci token                   */

{space}  ;
begin    Return (LEFT),
end      Return (RIGHT);
if       Return (IF),
then     Return (THEN),
{id}     Return (VAR),
{num}    Return (NUMBER),
{rel}    Return (RELATION),

%%
```

Specifikacija minijaturnog programskog jezika za program *Lex*

Slika 6

```
%{
   (* Biblioteke sistemskih potprograma *)
   Uses LexLib, YaccLib,
   (* Leksicki analizator *)
   {$I LexAna}
%}

/* Definisanje tokena */

%token if
%token then
%token left
%token right
%token number
%token var
%token relation

%%

/* Gramaticke pravila */

S     : if E then S
      | left S right
      |

E     . F R F

F       number | var

R     : relation

%%

BEGIN
   IF yyParse = 0 THEN
      WriteLn ('Rec je pravilna ')
   ELSE
      WriteLn ('Rec je nepravilna )
END
```

Specifikacija minijaturnog sintaksnog analizatora za program *Yacc*

vanje strukture jezika. Zadaje se nizom **produkcija** (pravila izvođenja) koje određuju gramatička pravila za pravljenje reči jezika, baš kao i u slučaju gramatike za prirodne jezike. Na primer, sledeća gramatička pravila opisuju jezik {**10**, **1100**, **111000**, ... }:

S → 1A0
A → 10
A → S

Prvo gramatičko pravilo „kaže" da reč **S** jezika mora imati cifru **1** na početku, **0** na kraju, a u sredini nešto što može zameniti promenljivu **A**. Drugo i treće pravilo objašnjavaju šta je to **A**: broj **10** ili (ponovo) reč jezika. Poslednje pravilo omogućava rekurziju i „pravljenje" beskonačnih reči. Pri zadavanju pravila razlikuju se **terminalni** i **neterminalni** simboli. Neterminalni simboli predstavljaju složene konstrukcije koje se dalje razlažu na jednostavnije; u suštini su to svojevrsne promenljive. Na primer, u prethodnoj gramatici simboli **S** i **A** su neterminalni. Terminalni simboli su oni simboli iz alfabeta koji se dalje ne razlažu. Pošto može postojati više produkcija oblika **A → B**, **A → C**, to se obično skraćeno piše **A → B | C**. Tako se poslednja dva pravila mogu skraćeno pisati kao **A → 10 | S**.

Prethodna gramatika opisuje jedan veoma jednostavan jezik. Reči jezika se obrazuju upotrebom gramatičkih pravila na sledeći način: pođe se od neterminalnog simbola koji je označen kao polazni – u našem slučaju to je **S**. Zatim se primenjuje proizvoljno izabrano gramatičko pravilo i na taj način se „razlažu"

Slika 5



Automat sa stekom

dobijeni neterminalni simboli. Postupak se ponavlja sve dok ne se ne izgube svi neterminali. Na primer:

S	**(Polazni simbol)**
1A0	**(Primena S → 1A0)**
1S0	**(Primena A → S)**
11A00	**(Ponovo primena S → 1A0)**
111000	**(Pravilo A → 10)**

Tako smo dobili reč **111000** sastavljenu samo od terminalnih simbola. Na sličan način se izvode i ostale reči. Sve reči koje se ovako mogu izvesti čine jedan jezik za koji se kaže da je zadat prethodnom gramatikom. Gramatike su uglavnom namenjene generisanju jezika (tj. opisivanju svih reči koje ulaze u jedan jezik), dok su automati namenjeni prepoznavanju reči iz tog jezika.

Gramatike su veoma moćno sredstvo za opisivanje jezika i njima se mogu opisati mnogo komplikovaniji skupovi nego konačnim automatima ili regularnim izrazima. To je zato što konačni automati imaju ograničenu „memoriju": oni naprosto nisu u stanju da opišu neke složenije veze između slova u reči. Konačni automati (odnosno regularni jezici) se najčešće koriste prilikom leksičke analize teksta – tada i nije potrebno neko dublje prepoznavanje konteksta. Tada se pojedine grupe slova tokenizuju. Međutim, prilikom sintaksne analize je to i te kako potrebno. Zato se tamo koriste **kontekstno slobodni** jezici. Naravno, kontekstno slobodni jezici „sadrže" regularne jezike: svaki regularni jezik je i kontekstno slobodan. To znači da se za svaki regularni jezik osim regularnog izraza i automata može napraviti i odgovarajuća gramatika koja opisuje taj jezik.

Kada se neki jezik opisuje gramatikom, uklanjaju se sve potencijalne višeznačnosti i nesporazumi oko značenja, što je naročito važno pri konstrukciji programskih jezika. One omogućuju kratak i veoma precizan opis velike klase jezika. Korišćenjem gramatika se zadaju **kontekstno-slobodni** jezici.

Kontekstno slobodne gramatike se koriste za opisivanje konstrukcija u programskim jezicima. Na primer, sledeća gramatika generiše sintaksno ispravne izraze:

S EOE | (S) | UE | LD
O / | | + | -
U – | –
L a | b | ... | z
D L | 0 | 1 | ... | 9
D

Kada se za jedan programski jezik definiše ovakva gramatika, tada je mnogo jednostavnije napisati prevodilac. Iz ovakvog zapisa se takođe mogu videti i neke nedoslednosti koje bi inače možda promakle kreatorima jezika.

AUTOMATI SA STEKOM

Kao što regularni jezici imaju svoje „mašine" koje ih prepoznaju (konačni automati), tako i za kontekstno slobodne jezike postoje odgovarajući automati: **automati sa stekom** (*pushdown automata*). Ovi automati, za razliku od konačnih, imaju i jednostavnu memoriju – stek u koji se mogu upisivati razni simboli. Na slici 5 prikazan je jedan takav automat. Automat redom čita simbole i na osnovu zadatog programa preduzima neku od tri moguće akcije: **PUSH c** – upisuje simbol **c** na stek, **POP** – skida simbol sa steka i **STATE s** – menja stanje automata. Automat se uvek nalazi u jednom od zadatih stanja, a program menja to stanje. Sa ulaza se redom čitaju simboli i automat menja stanja i simbol na vrhu steka kako mu program nalaže. Trenutnu konfiguraciju automata čine tekuće stanje, simbol na ulazu i simbol na vrhu steka. Od ova tri elementa zavisi koja će naredna akcija biti primenjena. Program je niz naredbi, a izvršava se uvek ona koja opisuje tekuću konfiguraciju. Sve naredbe su oblika

IF <**konfiguracija**> <**naredba**>

Naredba se izvršava samo ako konfiguracija **(s, x, y)** odgovara tekućem stanju: ako je tekuće stanje **S**, ako je ulazni simbol **x** i ako je na vrhu steka simbol **Y**.

Ako za primer uzmemo jezik svih reči oblika a^ncb^n, gde je broj a-ova jednak broju b-ova, program se može ovako napisati:

```
IF (s, a, )   PUSH a
IF (s, a, a)  PUSH a
IF (s, c, a)  STATE q
IF (q, b, a)  POP
```

Redosled naredbi u ovom programu nema značaja – automat uvek analizira sve naredbe i izvršava onu koja opisuje tekuće stanje. Prethodni program radi na sledeći način: na početku je stek prazan i automat upisuje simbol **a** sa ulaza na stek primenom prve naredbe. Zatim se redom čitaju svi simboli **a** i stavljaju na stek (primenjuje se druga naredba). Kada se naiđe na simbol koji razdvaja levu i desnu polovinu reči (**c**), automat primenjuje treću naredbu i menja stanje iz **S** u **Q** – to je interni signal da su na stek upisani svi simboli **a** i da sada treba čitati simbole **b**. Ideja je jednostavna – čitaju se redom simboli sa ulaza i za svaki pročitani se skida po jedan simbol **a** sa steka. Na taj način će na kraju stek biti prazan ako je reč ispravna (naredba broj 4).

Pisanje programa koji simulira rad ovakvog automata je jednostavan posao. I u opštem slučaju se stvar ne komplikuje previše: sve naredbe se jednostavno simuliraju, pa se može napraviti univerzalni automat koji radi na osnovu zadatog programa.

Jedan od veoma važnih delova prevodioca je i **sintaksni analizator**. To je program koji na osnovu ulaznog niza tokena i gramatičkih pravila prepoznaje programske konstrukcije (naredbe, deklaracije, ...), na osnovu čega se kasnije generiše odgovarajući kod. Slično kao što se leksička analiza obavlja korišćenjem konačnih automata, i sintaksna analiza se može uraditi pomoću automata sa stekom. I ovaj posao se može automatizovati: postoje generatori programa koji na osnovu gramatike generišu program koji prepoznaje sintaksne konstrukcije jezika. Jedan od najpoznatijih takvih programa je *Yacc*, koji specifikaciju pretvara u program koji simulira jednu vrstu automata sa stekom i na taj način prepoznaje konstrukcije programa. Na primer, za sledeću gramatiku

Slika 7



Tjuringova mašina

S if E then S
| left S right
E F R F
F number | var
R relation

specifikacija za *Yacc* (verzija za *Pascal*) prikazana je na slici 6. Terminale kao što su **number**, **var** itd. sintaksni analizator dobija od leksičkog analizatora kao izlazne tokene.

JEZICI BEZ OGRANIČENJA

Ni pomoću kontekstno-zavisnih jezika nije moguće opisati sve jezike koji su nam potrebni! Čak i tako sužen skup kao što je deo programskog jezika nije uvek kontekstno slobodan jezik. Primera radi, jezik $a^nb^nc^n$ = {**abc**, **aabbcc**, ...} ne može se opisati kontekstno slobodnom gramatikom. Stvar je u tome što kontekstno slobodni jezici (odnosno automati sa stekom) imaju ograničenu mogućnost da „pamte" informaciju o prethodno „obrađenim" simbolima. Za raspoznavanje ovakvih jezika potrebno je nešto još moćnije. Ta dodatna moć se dobija tako što se uvodi nova vrsta pravila izvođenja: sa leve strane se sada ne mora pojavljivati samo jedan neterminalni simbol, već proizvoljan niz terminalnih i neterminalnih simbola. Na primer, takva produkcija je

AsB → BaA

Izvođenje se obavlja slično kao i u prethodnim slučajevima. Ako sva pravila imaju manje simbola sa leve strane nego sa desne, dobija se tzv. kontekstno-osetljiv jezik. Ako se ne postavljaju takvi zahtevi, dobija se najšira klasa jezika (**jezici tipa 0** ili **jezici bez ograničenja**).

Na primer, jezik svih reči oblika $a^nb^nc^n$ može se generisati sledećom kontekstno-osetljivom gramatikom

S → ASBC | ABC
CB → BC
AB → ab
bB → bb
A → a
C → c

Prvo pravilo omogućava generisanje proizvoljnog niza oblika **AAABCBCBC**, pri čemu je broj simbola **A** jednak broju simbola **B** i **C**. Drugo pravilo omogućava da se svi simboli **B** „presele" na svoje mesto, iza simbola **A**. Tako se dobija niz **AAABBBCCC**. Sledeće pravilo omogućava da se simbol **B** iza simbola **A** pretvori u odgovarajući neterminal, a četvrto pravilo da se to lančano prenese na sve simbole **B**, ali samo ako su na odgovarajućem mestu. Primer generisanja jedne reči:

S
ASBC
AABCBC
AABCBC
AABBCC
AabBCC
aabBCC
aabbCC
aabbcC
aabbcc

TJURINGOVA MAŠINA

Kao i prethodne klase jezika, i jezici tipa 0 imaju mašinu koja ih raspoznaje: takav automat je legendarna **Tjuringova mašina**, za koju ste svakako već čuli. Umesto jednostavno organizovane memorije kao što je stek, Tjuringova mašina ima beskonačnu memoriju, sasvim slično organizovanu kao memorija standardnih računara. Ta memorija je predstavljena beskonačnom trakom koja je izdeljena na ćelije, a u svaku ćeliju se može upisivati po jedan simbol. Mašina pomoću glave može pristupati podacima na traci i može ih menjati (slika 7). Glava se može pomerati za po jednu poziciju levo i desno, može čitati ili upisivati simbol u tekuću ćeliju. Pri tome, slično kao kod automata sa stekom, Tjuringova mašina menja interno stanje. Rezultat rada mašine je niz simbola koji mašina ostavi u prvih nekoliko ćelija – u zavisnosti od toga kakav odgovor očekujemo. Primetite da je ovde povećana mogućnost da nam mašina saopšti neki komplikovaniji rezultat: kao rezultat prepoznavanja reči može se dobiti i odgovor „DA, TO JE REČ IZ JEZIKA", ako baš želimo.

Postoji više definicija Tjuringove mašine. Jedna od njih liči na definiciju automata sa stekom: na osnovu tekućeg simbola i stanja automat menja svoje stanje. Sve definicije se svode na postojanje nekoliko osnovnih operacija koje Tjuringova mašina može da obavlja: **LEFT** i **RIGHT** pomeraju glavu za jedno mesto, **WRITE c** za upisivanje simbola na tekuću poziciju, **IF c = s** za ispitivanje simbola na kome je glava i **GOTO** za promenu toka programa. Na primer, program koji u reči na traci zamenjuje sve simbole **A** simbolom **B**, sve do pojave tačke:

```
   if traka = '.' goto 2
1  if traka = 'A' write 'B'
   right
   goto 1
2  end
```

Ovakva definicija Tjuringove mašine je veoma „slobodna", ali u osnovi daje dobru sliku. Tjuringova mašina je u stvari generalizovan model savremenih računara i koristi se za matematičku definiciju algoritma. Ij. problema za koje se može napisati program, kaže se da je algoritam svaki program za Tjuringovu mašinu.

Tjuringova mašina se za prepoznavanje jezika može koristiti na sledeći način: na početku trake se upiše reč. Ako reč pripada jeziku, mašina će u prvoj ćeliji upisati simbol „1", a inače „0".

Tjuringova mašina ima mnogo veći teorijski nego praktičan značaj, za razliku od automata sa stekom, koji se u raznim varijantama obilno koriste za prepoznavanje kontekstno slobodnih jezika, što je jedan od centralnih zadataka svakog prevodioca.

POSLOVANJE PREDUZEĆA

Ovim tekstom započinjemo seriju o tome kako se piše softver za poslovanje jednog preduzeća. Cilj nam je da, u nekoliko nastavaka, obradimo najznačajnije detalje vezane za rad sa bazama podataka, tastaturom, ekranom i štampačima, postupno izgrađujući jednu kompletnu aplikaciju za vođenje *glavne knjige*.

Jovan Skuljan

Dosta je programera koji misle da je asembler podesan jedino za retka egzotične zahvate po „sistemskim dubinama" računara, gde ni jedan drugi jezik nije dovoljno efikasan. Međutim, uz jedan pravilno strukturan i sistematičan pristup, sa osloncem na gotove funkcije koje nudi DOS, čak i razvoj složenijeg mašinskog programa ne mora da predstavlja nikakav problem. Pogotovo ako je programer sa tim jezikom „odrastao".

Suštinu poslovanja nekog preduzeća čini, u najvećoj meri, novčana razmena sa drugim firmama, na osnovu kupovine i prodaje raznih dobara i usluga. Prema tome, sva preduzeća sa kojima poslujemo mogu se podeliti u dve osnovne grupe. Prvu grupu čine *kupci*, koji od nas kupuju nešto i za to nam plaćaju, a drugu čine *dobavljači*, od kojih mi nešto kupujemo i za to im plaćamo. Na primer, jedna zadruga koja posreduje u nalaženju poslova svojim članovima, ima kao kupce sva preduzeća kod kojih članovi zadruge rade. Zadruga tim preduzećima ispostavlja fakture na bruto iznos zarade, a preduzeća po tim fakturama vrše uplate. S druge strane, Elektrodistribucija, na primer, bi bila dobavljač, jer od nje zadruga kupuje struju. Ovoga puta dobavljač ispostavlja fakturu, a zadruga plaća.

Osnovu poslovanja predstavlja vođenje takozvanih *kartica kupaca* i *kartica dobavljača*. Radi se prosto o tome da se za svaku firmu sa kojom poslujemo mora voditi karton (kartica) na koju redovno upisujemo fakture (*račune*) i virmane (uplate) koji toj firmi odgovaraju. Primer jedne kartice kupca dat je na slici 1.

Recimo, ako kupac od nas hoće da kupi nešto za 1 000.00 dinara, mi mu ispostavljamo fakturu na taj iznos i upisujemo na njegovu karticu broj fakture, iznos, datum i slično. Kažemo da smo kupcu *uknjižili* fakturu. Sada čekamo da nam kupac plati tih 1 000.00 dinara i svako jutro proveravamo stanje svog računa u banci, prelistavajući pristigle virmane. Kad uplata stigne, mi na karticu kupca upisujemo broj virmana, iznos, datum i slično, a možemo odmah staviti i oznaku pored fakture da je naplaćena. Tada kažemo da su faktura i virman *povezani* (zatvoreni).

Kada unosimo u karticu dinarske iznose, moramo imati dve zasebne kolone, jednu za fakture, a drugu za virmane. Ne bi imalo mnogo smisla sve strpati u jednu kolonu, jer bi se pri sabiranju izgubila informacija koliko iznosi suma faktura, a koliko suma virmana, što je u knjigovodstvu veoma važno. Zato kartice i imaju dve kolone, pri čemu se jedna naziva *dugovna* i ima oznaku *DUGUJE*, a druga *potražna*, sa oznakom *POTRAŽUJE*. Oznake bi trebalo da budu jasne same po sebi. Iznos fakture ide na dugovnu stranu, jer kupac nama po toj fakturi duguje. Iznos virmana ide na potražnu stranu, jer je kupac nama uplatio i tako smanjio stanje svog računa, pa bi teoretski gledano mogao da potražuje nazad te pare od nas. Realno, kupac može da potražuje samo ako je greškom više uplatio, a i to se dešava.

Na potpuno sličan način kao kod kupaca možemo voditi i kartice dobavljača. Ovoga puta ćemo knjižiti fakture koje dobavljač nama ispostavlja i uplate koje mi dobavljaču šaljemo. Povezivanje faktura sa virmanima je isto kao i kod kupaca. Kartica opet ima dugovnu i potražnu stranu, s tim što sada fakture idu na potražnu (jer dobavljač od nas potražuje), a naše uplate na dugovnu stranu (jer dobavljač nama duguje, ako više uplatimo).

ANALITIKA I SINTETIKA

Vođenje kartica kupaca i dobavljača obično se označava kao *analitičko knjigovodstvo*, zato što se pri tome pojedinačno radi sa svakom firmom, svakom fakturom i svakim virmanom. Postoji sa druge strane i *sintetičko knjigovodstvo*, koje se bavi nekim drugim karticama na kojima se vodi poslovanje sa *svim* kupcima i *svim* dobavljačima zbirno. To je vođenje *glavne knjige*, o čemu ćemo uostalom kasnije detaljno govoriti.

Svaki programer će odmah uočiti da je vođenje kartica idealan posao za računar. Uzmimo, recimo, kartice kupaca. Koliko god da ima kupaca, mi svakom možemo dodeliti neku *šifru*, odnosno redni broj pod kojim će se on voditi u bazi podataka. Možemo, recimo, rezervisati na disku datoteku KUPCI.DAT, u kojoj će redom biti zapisani podaci o svakom kupcu (naziv, adresa sedišta, mesto, telefon, žiro račun i slično). Kartica tog kupca (ili bilo kog drugog) može biti organizovana u vidu dva nezavisna *niza* (lanca). Jedan niz bi bio niz faktura, a drugi niz virmana. Kada hoćemo da *uknjižimo* novu fakturu na karticu kupca, program treba samo da ubaci tu fakturu u postojeći lanac.

Kada bi nam bilo svejedno kojim redosledom se nižu fakture u lancu, svaka nova faktura bi mogla da se prikači na kraj lanca, što je i programski najjednostavnije izvesti. Međutim, daleko bolje bi bilo da nizovi faktura i virmana budu uređeni po rastućem datumu. Program za knjiženje (ulančavanje) nove fakture ili novog virmana bi zato morao da bude nešto komplikovaniji, jer nova faktura ili novi virman neće obavezno ići na kraj lanca (u knjigovodstvu se vrlo često vrši vraćanje na prošle datume).

Po potrebi, osim faktura i virmana, može se uvesti i lanac *ostalih knjiženja*, koja bi služila za raznorazne zahvate po karticama tipa *storniranja* (poništavanja) pogrešno knjiženih stavki, *preknjižavanja* (prenošenja) stavki sa jednog kupca na drugog i slično. Ko radi, taj i greši, pa su ovakvi zahvati na karticama uvek potrebni.

Štampanje kartice nekog kupca svelo bi se na paralelno prelistavanje sva tri lanca koji čine karticu (fakture, virmani, ostale stavke), po zadatoj šifri kupca. Pri štampanju bi se, naravno, sve stavke uredile po rastućim datumima, a na kraju bi se dao ukupan zbir dugovne i potražne strane, kao i *saldo* (razlika dugovne i potražne strane). Na primer, ako nam kupac duguje 15 000.00 dinara (toliko iznosi zbir svih faktura koje smo mu ispostavili), a potražuje 10 000.00 dinara (toliko nam je ukupno uplatio), onda je njegov saldo 5 000.00 dinara. Kada je saldo veći od nule, kupac nam duguje, a kada je manji od nule, kupac potražuje. U prvom slučaju ćemo se angažovati da od kupca, makar i sudskim putem, dobijemo pare koje nam duguje. U drugom slučaju obično nećemo baš biti tako hitri da vratimo pare. Možda ćemo se čak praviti kao da nemamo pojma i čekati da se kupac prvi javi... Ko kaže da knjigovodstvo nije zabavno.

GLAVNA KNJIGA

Kao što smo već rekli, glavna knjiga se odnosi na takozvano *sintetičko knjigovodstvo*, gde se poslovanje preduzeća prati zbirno. Kako to izgleda, pokazaćemo na primeru kupaca.

Formiraćemo jednu karticu na koju ćemo upisivati ukupan dnevni promet kupaca. Na primer, ako smo danas izdali dve stotine faktura raznim kupcima, sa ukupnim iznosom od 200 000.00 dinara, onda ćemo na pomenutu karticu upisati kao jednu stavku pod današnjim datumom iznos od 200 000.00 dinara na dugovnu stranu, sa obrazloženjem da je toliko fakturisano kupcima. Isto tako, ako nam je danas stiglo ukupno 150 000.00 dinara preko virmana koje šalju kupci, to ćemo upisati kao novu stavku na potražnoj strani (slika 2). Tako iz dana u dan možemo pratiti ukupan promet kupaca preko jedne jedine kartice.

Potpuno slično tome, dnevni promet dobavljača možemo upisivati na posebnu karticu. A neka treća kartica biće rezervisana za dnevne isplate i uplate na blagajni našeg preduzeća, i tako dalje. Svaka vrsta novčanog prometa imaće svoju posebnu karticu, a skup svih takvih kartica naziva se *glavna knjiga*.

Kao što kupci i dobavljači (odnosno njihove kartice) imaju svoje jedinstvene šifre, tako i karticama glavne knjige treba dodeliti odgovarajuće šifre. U ovom slučaju šifre imaju i poseban naziv: *konta*. Tako, na primer, kažemo da se ukupan promet svih kupaca vodi na kontu 12000, što u prevodu znači da ukupan promet kupaca vodimo na kartici broj 12000 u glavnoj knjizi. Slično tome, promet dobavljača ćemo voditi na kontu 22000 i tako dalje. Treba samo reći i to da konta ne možemo izmišljati po volji. Na primer, zakonom je propisano da se kupci moraju voditi na kontu 120, a dobavljači na 220. Naša sloboda je samo utoliko što možemo „razbiti" zadati trocifreni konto dopisujući još cifara. Obično se u praksi operiše petocifrenim ili šestocifrenim kontima. Tako, recimo, možemo uvesti konta 12000, 12001, 12002 i tako dalje, da bismo vodili pojedine aspekte poslovanja sa kupcima. Ali sva ta konta se odnose na kupce, pa zato i počinju sa 120.

Za nas, kao programere, svi ovi detalji oko kontiranja ne moraju imati većeg značaja osim informativnog. Mi ćemo svaki konto tretirati kao jedinstvenu šifru, a šta se pod tom šifrom krije važno je jedino onome ko se bavi knjigovodstvom. Nama je konto samo podatak potreban za pristup odgovarajućoj kartici, baš kao što je šifra kupca podatak za pristup njegovoj kartici.

Slika 1

RAČUNARI — Program za poslovanje preduzeća

KARTICE KUPACA

Naziv: IMPEX
Adresa: Bulevar Revolucije 17
Mesto: BEOGRAD

Šifra kupca: 2147
Žiro račun: 60805-621-457921

REDNI BROJ	FAKTURA BROJ	VIRMAN BROJ	OSTALO BROJ	DATUM DOKUMENTA	IZNOS DUGUJE	IZNOS POTRAŽUJE
1	12000			17.03.92	1.000,00	
2		14010		18.03.92		1.000,00
3	13000			02.03.92	9.000,00	
4	10656			07.03.92	5.000,00	
5		14000		08.03.92		8.000,00
				UKUPNO	15.000,00	10.000,00
				SALDO	5.000,00	

Kartice kupaca

Slika 2

RAČUNARI — Program za poslovanje preduzeća

GLAVNA KNJIGA

Opis: KUPCI — Broj konta: 12000

REDNI BROJ	STAVKA BROJ	DATUM STAVKE	IZNOS DUGUJE	IZNOS POTRAŽUJE
1	12321	14.03.92	200.000,00	
2	12002	14.03.92		150.000,00
		UKUPNO	200.000,00	150000,00
		SALDO	50.000,00	

Glavna knjiga

Slika 3

RAČUNARI — Program za poslovanje preduzeća

NALOG ZA KNJIŽENJE BROJ: 875
14.03.92.

REDNI BROJ	STAVKA BROJ	KONTO BROJ	OPIS STAVKE	DATUM STAVKE	IZNOS DUGUJE	IZNOS POTRAŽUJE
1	11053	12000	Fakturisano kupcima	14.03.92	200.000,00	
2	11054	75100	Neto zarada	14.03.92		180.000,00
3	11555	75101	Provizija	14.03.92		20000,00
				UKUPNO	200.000,00	200.000,00

Nalog za knjiženja

Slika 1

RAČUNARI — Program za poslovanje preduzeća

BRUTO BILANS

Broj konta	Početno stanje dugује	Početno stanje potražuje	Tekući promet duguje	Tekući promet potražuje	Ukupan promet duguje	Ukupan promet potražuje	Saldo duguje	Saldo potražuje
12000	5.000,00	1.000,00	12.000,00	10.000,00	17.000,00	11.000,00	6.000,00	0,00
120	5.000,00	1.000,00	12.000,00	10.000,00	17.000,00	11.000,00	6.000,00	0,00
75100	0,00	0,00	10000,00	1.000,00	10.000,00	7.000,00	3.000,00	0,00
75101	0,00	0,00	5.000,00	6.000,00	5.000,00	6.000,00	0,00	1.000,00
751	0,00	0,00	15000,00	13.000,00	15.000,00	13000,00	3.000,00	1000,00
1	5.000,00	1.000,00	12.000,00	10.000,00	17.000,00	11.000,00	6.000,00	0,00
7	0,00	0,00	15.000,00	13.000,00	15.000,00	13.000,00	3.000,00	1.000,00

Bruto balans

Vođenje glavne knjige u praksi se obavlja u nekoliko koraka. Pre svega formira se *nalog za knjiženje* koji sadrži popis svih stavki koje se za dati dan imaju upisati u kartice glavne knjige. Recimo, danas upisujemo samo ukupan iznos fakturisanja na konto 12000, od čega neto iznos na konto 75100, a proviziju na 75101. Nalog će imati tri stavke, kao na slici 3.

Zatim se pojedine stavke sa naloga upisuju na kartice odgovarajućih konta. Prva stavka ide na karticu 12000, druga na 75100, a treća na 75101. Kažemo da smo nalog *proknjižili* u glavnu knjigu.

Za razliku od kartica kupaca i dobavljača, svaka kartica glavne knjige se sastoji samo od jednog niza stavki, pri čemu stavka može imati i dugovnu i potražnu stranu. Stavke su takođe uređene po rastućem datumu. Program za štampanje kartice nekog konta treba samo da izlista odgovarajući lanac i na kraju da formira zbir dugovne i potražne strane sa izračunatim saldom.

ZAKLJUČNI LIST

Kada dođe vreme za takozvani *periodični obračun* (jednom u tri meseca), podaci iz glavne knjige se obrađuju i štampaju u vidu posebnog dokumenta koji se zove *zaključni list* ili *bruto bilans*. To je prosto zbirni popis stanja svih konta (svih kartica glavne knjige) zaključno sa danom obračuna (slika 4). Za svaki konto daju se četiri kolone. *Početno stanje* je preneto stanje konta iz prošle godine, *tekući promet* je ostvareni promet u tekućoj godini, *ukupan promet* je zbir početnog stanja i tekućeg prometa, a *saldo* je razlika dugovne i potražne strane ukupnog prometa. Ovoga puta i sam saldo ima dugovnu i potražnu stranu (duguje ako je pozitivan, a potražuje ako je negativan).

Sva konta koja počinju sa iste tri cifre čine *grupu*. Recimo, konta 12000, 12001 i tako dalje čine grupu 120. Zbirno stanje 120 daje se u jednom redu zaključnog lista odmah iza popisa konta koji tu grupu čine. I konačno, viši se takozvana *rekapitulacija*, gde se daju zbirna stanja po *klasama*, gde jednu klasu čine sva konta koja počinju istom cifrom. Dakle, postoji ukupno deset klasa sa oznakama od 0 do 9. Kupci će pripasti klasi 1, a dobavljači klasi 2.

PLANIRANJE SOFTVERA

Program za vođenje glavne knjige morao bi da ima sledeće mogućnosti:

1. Formiranje kontnog plana (unošenje opisa pojedinih konta i štampanje kontnog plana)
2. Formiranje naloga za knjiženja u glavnu knjigu
3. Pregled i ispravke naloga
4. Knjiženje naloga u glavnu knjigu
5. Kontrola knjiženja (da li je svaka stavka legla na karticu odgovarajućeg konta)
6. Pregled kartica (pojedinačni pregled na ekranu, štampanje jedne ili više kartica na štampaču, kao i štampanje zaključnog lista)

Pošto ćemo tokom ove serije tekstova detaljno opisati svaki od pomenutih poslova, sada se možemo zadržati samo na problemu baza podataka koje će nam za rad biti neophodne. Pre svega, jasno je da moramo imati datoteku STAVKE.DAT koja će sadržati stavke za knjiženje u glavnu knjigu. Svaka stavka može imati svoj jedinstveni identifikacioni broj, kao i podatke o tome na koji se konto odnosi, kom nalogu pripada, koliki su iznosi duguje i potražuje, i slično. Službenik koji radi na glavnoj knjizi ukucavaće stavke u fazi formiranja naloga, što bi moglo da se odvija po sledećem algoritmu:

1. Unesi broj naloga
2. Unesi broj prve slobodne stavke
3. Ukucaj opis i iznos stavke
4. Upiši stavku u bazu podataka
5. Pređi na sledeću stavku i ponovi ciklus od koraka 3

Može se lako obezbediti da posle svake upisane stavke brojač automatski bude uvećan za jedinicu, tako da korisnik o tome ne mora da brine. Posebnom komandom (tasterom ESC) završiće se formiranje naloga, a unete stavke nalaziće se u datoteci STAVKE.DAT.

DIREKTAN PRISTUP

Najjednostavnije je da svakoj stavci u datoteci odgovara jedan zapis, pri čemu su zapisi fiksne dužine (na primer 64 bajta) i ređaju se onako kako ide i identifikacioni (redni) broj stavke. Tako bi stavka broj 1 bila smeštena u prvom zapisu datoteke, a stavka broj 1000 u hiljaditom. Upis i čitanje bili bi maksimalno brzi, jer se pozicija svakog zapisa vrlo lako može računati na osnovu zadatog broja stavke i dužine zapisa.

Pretpostavimo da je za vođenje glavne knjige dovoljno 100000 stavki (najverovatnije da je i 10000 dovoljan broj, ali ne treba na takvim stvarima škrtariti). Pri dužini zapisa od 64 bajta, ukupna dužina datoteke bila bi 6400000 bajtova (oko 6 MB). Pitanje je samo da li treba unapred rezervisati toliki prostor ili pustiti da tokom rada datoteka po potrebi raste. Ukusi su tu različiti, ali naš savet je da se ipak ide na datoteke fiksne dužine, makar one bile i malo glomaznije, jer se tako sigurno održava dobra organizacija podataka na disku. Pri velikom broju datoteka koje dinamički menjaju svoju dužinu, vrlo brzo će doći do rasparčavanja i razbacivanja pojedinih delova iste datoteke po međusobno udaljenim delovima diska.

Organizacija datoteke na način koji smo upravo opisali ima smisla jedino ako unapred znamo da će svi zapisi biti iskorišćeni. Stavke glavne knjige taj uslov zadovoljavaju, jer nema razloga da neki brojevi stavki budu preskakani. Međutim, ako uzmemo primer konta, stvari će stajati drugačije. Svakako treba formirati datoteku KONTO.DAT koja će sadržati opise svih konta koja preduzeće u svom poslovanju koristi. Tako će, recimo, konto 12000 imati opis „Kupci", a konto 22000 opis „Dobavljači". Pošto

DATOTEKE POTREBNE ZA VOĐENJE GLAVNE KNJIGE

Broj datoteke: 01h

Naziv STAVKE DAT
Sadržaj Podaci o stavkama za knjiženje u glavnu knjigu
Duzina zapisa 64 bajta
Ukupan broj zapisa 100000

Polje	Ofset	Duzina	Tip	Opis
01h	00h	01h	NUM	Signal SLOBODNO/ZAUZETO
02h	01h	03h	NUM	Broj stavke (1 99999)
03h	04h	01h	NUM	Dan
04h	05h	01h	NUM	Mesec
05h	06h	01h	NUM	Godina
06h	07h	02h	NUM	Broj naloga za knjiženje
07h	09h	20h	TXT	Opis knjizenja
08h	29h	0Ah	NUM	Iznos duguje
09h	33h	0Ah	NUM	Iznos potražuje
0Ah	3Dh	03h	NUM	Broj konta na koji se stavka knjiži

Broj datoteke: 02h

Naziv NALOG DAT
Sadržaj Podaci o datumu i tipu naloga za knjizenje
Duzina zapisa 8 bajtova
Ukupan broj zapisa 10000

Polje	Ofset	Dužina	Tip	Opis
01h	00h	01h	NUM	Signal SLOBODNO/ZAUZETO
02h	01h	02h	NUM	Broj naloga (1-9999)
03h	03h	01h	NUM	Dan
04h	04h	01h	NUM	Mesec
05h	05h	01h	NUM	Godina.
06h	06h	01h	NUM	Tip naloga

Broj datoteke: 03h

Naziv NALOGDUG DAT
Sadržaj Podaci o broju stavki i dugovnom iznosu naloga
Duzina zapisa 16 bajtova
Ukupan broj zapisa 10000

Polja	Ofset	Dužina	Tip	Opis
01h	00h	01h	NUM	Signal SLOBODNO/ZAUZETO
02h	01h	02h	NUM	Ukupan broj stavki
03h	03h	0Ah	NUM	Ukupan iznos duguje

Broj datoteke: 04h

Naziv NALOGPOT DAT
Sadrzaj Podac o broju stavki i potražnom iznosu naloga
Duzina zapisa 16 bajtova
Ukupan broj zapisa 10000

Polja	Ofset	Duzina	Tip	Opis
01h	00h	01h	NUM	Signal SLOBODNO/ZAUZETO
02h	01h	02h	NUM	Ukupan broj stavki
03h	03h	0Ah	NUM	Ukupan iznos potražuje

Broj datoteke: 05h

Naziv TIPNAL DAT
Sadržaj Opis tipa naloga
Duzina zapisa 32 bajta
Ukupan broj zapisa 10

Polja	Ofset	Dužina	Tip	Opis
01h	00h	01h	NUM	Signal SLOBODNO/ZAUZETO
02h	01h	01h	NUM	Oznaka tipa naloga (1 9)
03h	02h	01h	NUM	Dan
04h	03h	01h	NUM	Mesec
05h	04h	01h	NUM	Godina
06h	05h	19h	TXT	Opis tipa naloga

Broj datoteke: 06h

Naziv NALSTAV GEN
Sadržaj Generator lanca stavki koje pripadaju jednom nalogu
Dužina zapisa 8 bajtova
Ukupan broj zapisa 10000

Polja	Ofset	Dužina	Tip	Opis
01h	00h	01h	NUM	Signal SLOBODNO/ZAUZETO
02h	01h	03h	NUM	Broj prve stavke u lancu
03h	04h	03h	NUM	Broj poslednje stavke u lancu

Broj datoteke: 07h

Naziv NALSTAV LAN
Sadrzaj Lanac stavki koje pripadaju jednom nalogu
Duzina zapisa 16 bajtova
Ukupan broj zapisa 100000

Polje	Ofset	Duzina	Tip	Opis
01h	00h	01h	NUM	Signal SLOBODNO/ZAUZETO
02h	01h	03h	NUM	Broj naloga kome stavka pripada
03h	04h	03h	NUM	Broj prethodne stavke u lancu
04h	07h	03h	NUM	Broj sledece stavke u lancu

Broj datoteke: 08h

Naziv KONTSTAV GEN
Sadrzaj Generator lanca stavki koje pripadaju jednom kontu
Duzina zapisa 8 bajtova
Ukupan broj zapisa 10000

Polje	Ofset	Duzina	Tip	Opis
01h	00h	01h	NUM	Signal SLOBODNO/ZAUZETO
02h	01h	03h	NUM	Broj prve stavke u lancu
03h	04h	03h	NUM	Broj poslednje stavke u lancu

Broj datoteke: 09h

Naziv KONTSTAV LAN
Sadrzaj Lanac stavki koje pripadaju jednom kontu
Duzina zapisa 16 bajtova
Ukupan broj zapisa 100000

Polja	Ofset	Duzina	Tip	Opis
01h	00h	01h	NUM	Signal SLOBODNO/ZAUZETO
02h	01h	03h	NUM	Broj konta kome stavka pripada
03h	04h	03h	NUM	Broj prethodne stavke u lancu
04h	07h	03h	NUM	Broj sledeće stavke u lancu

Broj datoteke: 0Ah

Naziv KONTO DAT
Sadrzaj Opisi pojedinih konta koji ulaze u glavnu knjigu
Dužina zapisa 80 bajtova
Ukupan broj zapisa 10000

Polje	Ofset	Duzina	Tip	Opis
01h	00h	01h	NUM	Signal SLOBODNO/ZAUZETO
02h	01h	03h	NUM	Broj konta (00001 99999)
03h	04h	01h	NUM	Dan
04h	05h	01h	NUM	Mesec.
05h	06h	01h	NUM	Godina
06h	07h	24h	TXT	Opis konta (prvi red)
07h	2Bh	24h	TXT	Opis konta (drugi red)

Broj datoteke: 0Bh

Naziv KONTO × DAT
Sadrzaj Indeks za pristup datoteci sa opisom konta
Dužina zapisa 4 bajta
Ukupan broj zapisa 100000

Polje	Ofset	Duzina	Tip	Opis
01h	00h	01h	NUM	Signal SLOBODNO/ZAUZETO
02h	01h	03h	NUM	Indeks za pristup datoteci sa opisom konta.

Broj datoteke: 0Ch

Naziv KONTOPTR DAT
Sadrzaj Pokazivač slobodnih zapisa u datoteci KONTO DAT
Duzina zapisa 8 bajtova
Ukupan broj zapisa 2

Polje	Ofset	Dužina	Tip	Opis
01h	00h	01h	NUM	Signal SLOBODNO/ZAUZETO
02h	01h	03h	NUM	Broj prvog slobodnog zapisa
03h	04h	03h	NUM	Broj poslednjeg slobodnog zapisa

je broj konta pailociran, opet bi nam odgovarala datoteka sa 100000 zapisa, ali bi sada najveći deo tog prostora ostao neiskorišćen. Činjenica je, naime, da se u poslovanju bilo koje firme ne koristi više od nekoliko hiljada konta.

Očigledno, datoteku KONTO.DAT nema smisla organizovati po principu direktnog pristupa. Ona treba da ima samo nekoliko hiljada zapisa, pri čemu pozicija zapisa neće odgovarati broju konta. Kako onda pronaći brzo neki zapis po zadatom broju konta? Pretraživanje, naravno, treba odmah odbaciti kao beznadežno sporo. Daleko je efikasniji metod *indeksiranja*.

INDEKSIRANJE

Uvešćemo zato jednu datoteku pod nazivom KONTO#.DAT i zvaćemo je *indeksnom datotekom*. Ona zapravo ima strukturu obične datoteke sa direktnim pristupom, pri čemu svaki njen zapis daje informaciju o tome gde se u datoteci KONTO.DAT nalazi opis zadatog konta. To znači da zapisi indeksne datoteke mogu biti minimalne dužine (recimo tri bajta) taman koliko je potrebno da se sačuva indeks u rasponu 1-100000.

Korišćenjem metoda indeksiranja, pristup datoteci KONTO.DAT vrši se u dva koraka. Prvo se sa zadatim brojem konta uzme iz KONTO#.DAT *indeks* koji tom kontu odgovara, a zatim se sa tim indeksom pristupa opisu konta u KONTO.DAT. Očigledno, opisi konta se ne moraju ređati po nekom redosledu. Može, na primer, prvi opis u datoteci KONTO.DAT da odgovara kontu 12000. Međutim, u tom slučaju, u datoteci KONTO#.DAT, zapis sa rednim brojem 12000 mora sadržati indeks 1.

Naravno, i dalje ostaje činjenica da će najveći deo datoteke KONTO#.DAT ostati neiskorišćen, ali je gubitak ovoga puta zaista više simboličan. Cela datoteka KONTO#.DAT, pri dužini zapisa od 3 bajta, zauzima svega 300000 bajtova.

ULANČANI ZAPISI

Preostaje još da vidimo na koji način ćemo na disku čuvati podatke koji čine jedan lanac ili niz, kao što je to slučaj sa *nalogom* za knjiženje ili sa *karticom* zadatog konta. I nalog i kartica se sastoje od niza pojedinačnih stavki za knjiženje. Pošto se svi podaci o stavkama već čuvaju u datoteci STAVKE.DAT, nalog ili kartica će potpuno biti određeni ako se samo navede popis svih *brojeva* stavki koje ulaze u odgovarajući lanac.

Pozabavimo se prvo nalozima za knjiženje. Uvešćemo jednu datoteku pod nazivom NALSTAV.LAN, koja predstavlja *lanac stavki* koje sačinjavaju *nalog*. Datoteka je obična, sa direktnim pristupom, i ima onoliko zapisa koliko ima i stavki u datoteci STAVKE.DAT (dogovorili smo se da to bude 100000). Svaki zapis LAN datoteke sadrži samo tri podatka:

1. Kom lancu stavka pripada (kom nalogu)
2. Koji je prethodni element lanca
3. Koji je sledeći element lanca

Ako je neka stavka prva u lancu, onda na mestu oznake za prethodni element stoji nula. Isto tako, ako je stavka poslednja u lancu, na mestu oznake za sledeći element stoji nula. Stavka koja je jedina u lancu ima oba pokazivača nule.

Osim toga, uvešćemo i datoteku NALSTAV.GEN, koju ćemo zvati *generatorom lanca*. Ona ima onoliko zapisa koliko predviđamo da će biti naloga (recimo 10000) i organizovana je na principu direktnog pristupa. Svaki zapis sadrži dve informacije:

1. Koja je prva stavka u lancu
2. Koja je poslednja stavka u lancu

Na taj način, kombinacijom jedne GEN i jedne LAN datoteke imamo potpunu informaciju o bilo kom lancu stavki. Recimo da nas interesuje nalog broj 4001. U datoteci NALSTAV.GEN pročitaćemo da nalog počinje stavkom 7245, a završava se stavkom 7301. Zatim prelazimo u datoteku NALSTAV.LAN i pratimo lanac od prve stavke unapred ili od poslednje unazad.

Podesno bi bilo da uvedemo još nekoliko datoteka koje opisuju dati nalog. Pre svega, datoteka NALOG.DAT može da daje informaciju o datumu naloga i tipu naloga (da li je to početno stanje, izlazne fakture, ulazne fakture i slično). Osim toga, datoteke NALOGDUG.DAT i NALOGPOT.DAT mogu da sadrže ukupan zbir dugovne, odnosno potražne strane naloga, sa informacijom o tome koliko stavki nalog sadrži. Moguće je, naravno, sve ovo obuhvatiti jednom datotekom, ali mi smo se opredelili da stvar razbijemo na manje celine.

Lanac stavki koje čine jednu karticu definisaćemo parom datoteka KONTSTAV.GEN i KONTSTAV.LAN. Slično kao u slučaju naloga, ovde GEN datoteka daje prvu i poslednju stavku u lancu, a LAN datoteka predstavlja sam lanac.

PLANIRANJE BAZE

Preostaje da se, na kraju, opredelimo za konkretan format i veličinu datoteka, kako bismo pripremili teren za pisanje programa. Za vođenje glavne knjige biće nam potrebno dvanaest datoteka, koje detaljno opisujemo u priloženoj tabeli. Međutim, posebno ćemo izneti i neke opšte karakteristike.

Svaka datoteka je sastavljena od zapisa fiksne dužine, pri čemu numeracija zapisa ide od *nule*. Prvi po redu je, dakle, *nulti* zapis, koji će uvek biti prazan, i služiće kao uzorak nepopunjenog zapisa. Ovo će biti zgodno kada budemo pisali program za brisanje zadatog zapisa. Posao će se svesti prosto na kopiranje nultog zapisa na zadato mesto.

Prisustvo nultog zapisa znači i to da od, recimo, 100000 postojećih zapisa samo njih 99999 mogu da se iskoriste za čuvanje podataka, o čemu stalno moramo voditi računa.

Svaki zapis se sastoji od pojedinačnih *polja* zadate dužine i tipa. Neka polja biće rezervisana za čuvanje *numeričkih* podataka (NUM), a neka za *tekst* (TXT). Ako je polje numeričkog tipa, a prazno je, sadržaće nule. Međutim, prazno polje tekstualnog tipa će biti popunjeno blanko simbolima (20H).

Polja su označena rednim brojevima, počev od jedinice. Prvo polje u bilo kom zapisu bilo koje datoteke *uvek* je dužine jedan bajt i numeričkog je tipa. To polje sadrži 00H ako je zapis slobodan za čitanje, a FFH ako je zauzet. Situacija da zapis bude zauzet za čitanje može se desiti, naravno, samo ako više korisnika u mreži rade sa istom bazom podataka u trenutku dok jedan od njih upravo upisuje podatke u zadati zapis.

Sva *numerička* polja su u suštini celobrojnog tipa. Dinarski iznosi se čuvaju pomnoženi sa sto. Numerički podatak se upisuje tako da prvo ide bajt najniže vrednosti, a na kraju bajt najviše vrednosti.

Tekstualna polja predstavljaju nizove karaktera onako kako su uneti, bajt po bajt.

Nastavak sa strane 13

ne prelaze ukupnu memoriju (jedan draft karakter zahteva 30 bajtova, a jedan LQ karakter 114 bajtova). Pored toga, opciono se može nabaviti i kartridž sa dodatnim fontovima (Bold PS, Roman, Prestige, Sans Serif, Orator, Script i OCR-B).

...I IZVANREDNA GRAFIKA

Ako se za tekstualni otisak može reći da je veoma dobar, onda za grafiku ostaje samo jedan pridev – izvanredna. Ovo je, uz Epson LQ 1060, jedini pravi matrični štampač (ne računajući hibride kakvi su ink-jet i bubble-jet štampači) sa kojim sam do sada imao posla a da se nije ni malo namučio sa test slikom. Svih osam nijansi sivog, uz potpuno crni i potpuno beli uzorak, KX-P1450 je izvukao potpuno jasno i bez i najmanje problema, a krugovi su zaista krugovi. Čak i stepenu zacrnjenja nema šta da se prigovori, uprkos previše nežnoj traci i uzimajući u obzir činjenicu da je ipak reč o matričnom štampaču. Na raspolaganju su 8-pinska grafika sa maksimalnom vertikalnom rezolucijom od 240 dpi i 24-pinska grafika sa maksimalnom vertikalnom rezolucijom od 360 dpi. Horizontalna rezolucija u oba slučaja iznosi 180 dpi. Znajući da se ovakva rezolucije postižu pre svega pomeranjem papira za mala iznose, što dovodi do preklapanja tačkica, sasvim je jasno da sjajan otisak ide na adresu robusne i kvalitetne mehanike.

POUZDAN, IZDRŽLJIV, BRZ

Da bi se izveo kratak zaključak o nekom hardverskom proizvodu, neophodno je sabrati sve mane i sve vrline. U ovom slučaju je zbir vrlo neuravnotežen – mana je malo, a vrlina mnogo više. Najveća mana ovog inače vrlo uravnoteženog modela je svakako kontrolni panel, koji daleko zaostaje za mehanikom i preciznošću ostalih delova i deluje nekako nedovršeno, kao stvar koje su se projektanti setili tek naknadno i na brzinu ubacili u već završen projekat. Pored toga, izvesne zamerke se mogu staviti i na malu i krhku kasetu sa trakom, nežne poluge za uvlačenje papira i preklapanje između pojedinačnih listova i beskonačnog papira, kao i na relativno zastareli (mada sasvim efikasan) mehanizam za pritiskanje papira na valjak. Ako bih hteo da budem sitničav, a moram budući da se radi o testu, morao bih da spomenem i plastiku (mada u svim drugim aspektima izvanredno rešenu) vođicu pojedinačnih listova. Izuzev kontrolnog panela, reč je uglavnom o sitnim manama koje bi se pre mogle smatrati kozmetičkim nego funkcionalnim.

Dobre strane su znatno izraženije i ogledaju se pre svega u veoma čvrstoj i robusnoj konstrukciji, preciznom rukovanju papirom, ubedljivom tekstualnom i zaista izvanrednom grafičkom otisku i veoma solidnom uputstvu. Sve u svemu, „Panasonic" KX-P1540 je štampač koji pretenduje na gornju srednju klasu, kako po imenu, tako i po postignutim performansama. Uz ukupnu cenu od oko 2 000 DEM, što predstavlja neki realni prosek na našem tržištu za ovu klasu uređaja, ovaj štampač predstavlja dobar izbor za sve one kojima je potreban pouzdan, izdržljiv i relativno brz štampač A3 formata sa kvalitetnim otiskom i dobrom podrškom. Razmislite o njemu.

Korisna adresa

D.P. IBIS-SYS
11070 Beograd, Pariske komune 22
Tel. 011/609-650, fax 011/699-081

PROGRAMIRANJE LOKALNIH MREŽA

Ako ste o programiranju „jednog" računara već naučili sve što se naučiti može, možda je vreme da pređete na programiranje lokalnih mreža. Nova oblast će vam, uz mnogo zadovoljstva, doneti i saznanje da se sopstvenim aplikacijama lako mogu prevazići neka tipična ograničenja mrežnih operativnih aistema.

Ljubomir Josifovski

Uvreženo je mišljenje da su deljene datoteke na zajedničkom (mrežnom) disku lokalne mreže jedini način za prenos podataka između mrežnih stanica. Ipak postoji i bolji metod za slanje i prijem podataka i poruka između stanica, koji nas oslobađa briga oko slobodnog mesta na disku, prava pristupa direktorijumu i zaključavanja deljenih datoteka. Koristeći usluge NetBIOS-a možete slobodno da zaboravite na ove i ostale mrežne zvrčke. Programiranje NetBIOS-a je dovoljno jednostavno da se upustite u kreiranje sopstvenih aplikacija koje prevazilaze ograničenja mrežnih operativnih sistema.

Koje vam usluge NetBIOS pruža? Pre svega, razmenu poruka između mrežnih stanica putem slanja poruka, pri čemu „poruka" može da bude bilo šta dužine do najviše 64K. Naravno, niko vam ne brani da pošaljete 2, 3 ili koliko već želite poruka ako vam je 64 K nedovoljno.

NETBIOS I NJEGOVE USLUGE

Malo formalnije, NetBIOS (*Network Basic Input/Output System*) je API (*Aplication Program Interface*) za razmenu podataka. Prema ISO/OSI referentnom modelu, predstavljao bi interfejs ka sesijskom sloju. Takođe realizuje protokole za prenos u trećem, četvrtom i petom sloju (prema istom referentnom modelu). NetBIOS emulaciju na Novell-u dobijate startovanjem programa netbios.exe i int2f.com. Nije loše i da malo podesite parametre emulatora (ima ih desetak, a podešavanja se upisuju u shell.cfg fajl).

Komande NetBIOS-u zadajete tako što najpre popunite specijalnu strukturu zvanu NCB (*NetBIOS Controll Block*) potrebnim podacima, podesite ES:BX da pokazuje na početak NCB-a, a zatim (nimalo originalno) uradite jedan Int 5Ch. Struktura NCB-a data je na slici 1, a primer kako se NetBIOS poziva na slici 2.

Popunjavanje strukture je jednostavno. U polje **command** stavljate kod naredbe, a u ostala polja kontrolnog bloka njene parametre. Pre nego što počnete da popunjavate NCB, popunite celu strukturu nulama – uštedećete sebi glavobolje oko programa koji čas rade, a čas ne rade (slika 3). Ako imate jakog razloga da to ne uradite, obavezno upišite 0 bar u polje **lananum**. Nakon što NetBIOS odradi svoje, u polju **retcode** vraća kod koji pokazuje uspešnost zahtevane operacije – 0 znači da je sve prošlo OK, a sve ostalo je kod greške. Izuzetak je kod 255, koji dobijate kada zahtevate da se komanda izvrši bez čekanja, tj. da se kontrola odmah vrati vašem programu. U tom slučaju vaš program nastavlja sa radom odmah, ne čekajući da se zadata komanda izvrši do kraja. NetBIOS će pozvati vašu interapt rutinu kada kompletira zadatu komandu i kod greške smestiti u polje **cmdcplt** (0 ako je sve prošlo OK). Adresu vaše rutine koja će biti pozvana zadajete takođe preko NCB-a (prilikom izdavanja komande), i to segment u polju **postrtnsegment**, ofset u **postrtnoffset**. U terminologiji NetBIOS-a ove se komande zovu komande sa **No-Wait** opcijom, a vaša interapt rutina **post** rutina (u C-u je obavezno deklarisati kao **interrupt**).

Na slici 4 pobrojane su neke od važnijih komandi, njihovi kodovi (kod komande sa **No-Wait** opcijom dobija se tako što kodu komande dodamo 80h) i polja u kojima se prenose parametri (pored koda naredbe u polju **command**, povratnog koda u polju **retcode** i **cmdcplt** ako eventualno koristimo **No-Wait** opciju).

Komande su podeljene u 4 grupe: rad sa imenima, rad sa datagramima, rad sa sesijama i ostalo (režijske komande).

RAD SA IMENIMA

Ime je niz od najviše 16 znakova (razlikuju se mala od velikih slova), pri čemu je IBM za sebe rezervisao određena imena. Zdravije je da koristite samo prvih 15 slova, prvo slovo ne sme da bude *, a prva tri ni u snu **IBM**. Ime može da bude jedinstveno (dodaje se komandom **Add Name**) ili grupno (dodaje se komandom **Add Group Name**). Dodavanje imena nazivamo registracija imena. Jedinstveno ime je jedinstveno ne samo za jednu stanicu, nego za čitavu mrežu. Pokušaj dodavanja identičnog imena na vašoj ili drugoj stanici izazvaće grešku. Grupno ime može biti registrovano proizvoljan broj puta, na istoj ili različitim mrežnim stanicama.

Više korisnika mogu primati poruke koristeći isto (grupno) ime. Nakon što registruje ime, NetBIOS vraća njegov broj, koji koristite u daljnim operacijama umesto da stalno navodite ime. Imena se čuvaju u lokalnoj tabeli u memoriji mrežne stanice. Broj 1 odgovara permanentnom imenu stanice (**Permanent node name**, **Permanent node adress**), koje je 6 znakova dugačko i koje je garantovano jedinstveno za svaku mrežnu kartu. Preporuka je da ne ovo permanentno ime stanica zaboravite i ga ne koristite u svojim programima. U suprotnom, može da se desi da vaši programi prestanu da rade onog trenutka kada serviser zameni neispravnu karticu novom. Na slici 5 dat je jednostavan primer kako se registruje ime. Ime deregistrirate sa **Delete Name** komandom. Pri tome se u polje **name** stavlja ime koje želite da obrišete iz lokalne tabele.

SLANJE DATAGRAMA

Datagrami (= telegrami) su poruke za koje ne postoji potvrda prijema niti kontrola grešaka. Ako primalac ne postoji, ili nije izdao komandu za prijem datagrama, ili iz nekog drugog razloga nije u mogućnosti da primi poslati datagram, pošaljilac datagrama ne dobija nikakvu povratnu informaciju o tome. Dakle, pogodni su za prenos kratkih poruka. Da biste ne-

Listing 1

```
typedef struct {
   unsigned char command,       /* command code                              */
   unsigned char retcode        /* immediate return code                     */
   unsigned char lsn,           /* logical session number                    */
   unsigned char num,           /* name number                               */
   unsigned int bufferoffset,   /* buffer adress, segmentbuffset             */
   unsigned int buffersegment,
   unsigned int lenght,         /* buffer lenght                             */
   char callname[16],           /* remote name                               */
   char name[16],               /* my name                                   */
   unsigned char rto,           /* receive time out                          */
   unsigned char sto,           /* send time out                             */
   unsigned int postrtbuffset,  /* post routine adress, segmentbuffset       */
   unsigned int postrtnsegment,
   unsigned char lananum,       /* lan adpter number, put 0                  */
   unsigned char cmdcplt,       /* finall return code                        */
   char reservedarea[14];
} NCB;

NCB myncb,
```

Struktura NCB-a

Listing 2

```
void request (void)
{
   _ES = (unsigned int) FP_SEG( &myncb );
   _BX = (unsigned int) FP_OFF( &myncb ),
   geninterrupt (0x5c),
}
```

Poziv NetBIOS-a

Listing 3

```
void zeroit (void)
{
   int i,
   char *a;

   a = (char *)(&myncb),
   for (i=0, i < (sizeof (NCB) / sizeof (char)), i++, a++)
      *a = '\0',
}
```

Popunjavanje NCB-a nulama

Listing 4

```
Add name           30h, B0h (NoWait( In  name, post(?(
                                         Out  num
Add group name     36h, B6h (NoWait( In  name, post(?(
                                         Out  num
Delete name        31h, B1h (NoWait( In  name, post(?(
                                         Out  /

Send datagram      20h, A0h (NoWait( In  num, buffer, lenght, callname, post(?(
                                         Out  /
Receive datagram   21h  A1h (NoWait( In  num, buffer, lenght, post(?(
                                         Out  lenght, callname

Call               10h, 90h (NoWait( In  callname, name, rto, sto, post(?)
                                         Out  lsn
Listen             11h, 91h (NoWait( In  callname  name, rto, sto, post(?(
                                         Out  lsn
Send               14h, 94h (NoWait( In  lsn, buffer, lenght, post(?(
                                         Out  /
Receive            15h, 95h (NoWait( In  lsn, buffer, lenght, post(?(
                                         Out  lenght
Session Status     34h  B4h (NoWait( In  buffer, lenght, name, post(?(
                                         Out  num
Hang Up            12h, 92h (NoWait( In  lsn, post(?(
                                         Out  /

Adapter Status     33h  B3h (NoWait( In  buffer  lenght, callname, post(?(
                                         Out  lenght
```

Važnija komande NetBIOS-a, njihovi kodovi i parametri

Listing 5

```
void AddName (char *name(
{
   int i,

   zeroit ((,

   myncb command - ADD_NAME
   for (i = 0  (i < 16( && *name  i++, name++(
      myncb name[i( = *name

   request ((,

   if (myncb retcode == 0(
      printf ("&& Name num - &&", (int(myncb num(,
   else
      printf ("Error no  &&", (int(myncb retcode(,
}
```

Dodavanje imena u lokalnu tabelu

Listing 6

```
01h   Invalid buffer lenght
03h   Invalid command
05h   Command timedout
06h   Incomplete received message
08h   Invalid local session number
0Ah   Session has been closed
0Bh   Command was canceled
0Dh   Duplicate name in local NetBIOS table
0Eh   NetBIOS table full
0Fh   Name has active sessions and was is deregistrated
11h   NetBIOS local session table full
12h   Session open rejected because no Listen is outstanding
13h   Illegal name number
14h   Cannot find name called or no answer
15h   Name not found, or cannot specify  *  or 00h as first byte of name,
      or the name is deregistrated and cannot be used
16h   Name is in use on remote adapter
17h   Name is deleted
18h   Session ended abnormally
19h   Name conflict detected
24h   Command completed while cancel occuring
0?h   Reserved name specified for Add Group Name
0?h   Command not valid to cancel
```

Kodovi češćih grešaka

Listing 7

```
void receivewait (void)
{
   zeroit ((,

   myncb command = RECEIVE_DATAGRAM
   myncb num = receivernamenumber
   myncb Buffersegment = FP_SEG ( receivebuffer (,
   myncb Bufferoffset = FP_OFF ( receivebuffer (,
   myncb lenght = MAXMESSAGELENGHT

   request (),
}
```

Prijem datagrama

Listing 7a

```
void sendwait (void)
{
   zeroit ((,

   myncb command - SEND_DATAGRAM
   myncb.num = callernamenumber,
   myncb bufferoffset - FP_OFF( sendbuffer (,
   myncb buffersegment = FP_SEG( sendbuffer (
   myncb lenght = strlen (sendbuffer(
   for (i = 0, (i < 16( && *receivername, i++, receivername++(
      myncb.callname[i( = *receivername,

   request ((,
}
```

Slanje datagrama

Listing 8

```
typedef struct
{
   char name[16(,
   unsigned char namenum,
   unsigned char namestatus,

} NAMEENTRY,

typedef struct
{
   unsigned char permanentnodename[6(,
   unsigned char majorversionnumber
   unsigned char alwayszero,
   unsigned char lanadaptertype,
   unsigned char minorversionnumber;

   unsigned int reportingperiodminutes
   unsigned int framerejectedreceivecount,
   unsigned int framerejectedsentcount,
   unsigned int t_framereceiveerrorcount,
   unsigned int xmitabortcount
   unsigned long successfultransmitcount,
   unsigned long successfulreceivecount,
   unsigned int t_transmiterrorcount,
   unsigned int retrxmitbufferdepletioncount,
   unsigned int expiredt1timercount,
   unsigned int expiredtitimercount,   -

   DUMMY far *localstatp,

   unsigned int freecommandblocks,
   unsigned int currentcommands
   unsigned int maximumcommands
   unsigned int transmitbufferdepletioncount,
   unsigned int maximumdatagrampacketsize,
   unsigned int pendingsessioncount,
   unsigned int maxpendingsessioncount
   unsigned int maximumsessions
   unsigned int maximumsessionpacketsize,
   unsigned int nametableentrycount

   NAMEENTRY tableentry[MAX_NAMES(,

} STATUS;

STATUS mystat
```

Struktura polja kojeg popunjava *Adapter Status* komanda

Listing 9

```
switch (mystat tableentry[i( namestatus & 0x07(
   {
   case 0x00  printf ("Name is in registration process  "(
              break,
   case 0x04  printf ("Name is registered, "),
              break,
   case 0x05  printf ("Name is deregistered, "(
              break,
   case 0x06: printf ("Name is a detected duplicate name  "(
              break,
   case 0x07  printf ("Name is a detected duplicate name that is pending for \
                    deregistration  "(
   {

if (mystat tableentry[i( namestatus & 0x80(
   printf ("GROUP name \n"(,
else
   printf ("UNIQUE name \n"(,
```

Značenje polja *namestatus*



kome poslali datagram potrebno je da znate njegovo ime. Da bi primalac primio datagram mora da izda **Receive Datagram** komandu pre nego što pošaljilac izda **Send Datagram** komandu. Kod **Send Datagram** komande u polja **buffersegment** i **bufferoffset** upišite segment i ofset adrese gde je smeštena poruka (datagram) koji šaljete, u polje **lenght** dužinu poruke, u polje **num** broj imena u lokalnoj tabeli koji ste prethodno dobili prilikom registracije, a u polje **callname** ime primaoca. Kod **Receive Datagram** komande popunjava se polje **num** brojem vašeg imena, dok u **buffersegment**, **bufferofset** i **lenght** stavljate segment i ofset adrese i dužinu bafera gde će poruka biti smeštena nakon prijema. Ako je poruka uspešno primljena, nakon kompletiranja ove komande u polje **callname** biće ime pošaljioca, a u **lenght** dužina primljene poruke. Oprez! Ako je dužina bafera koji odvajate za prijem poruke manja od dužine poruke, prijem će biti neuspešan. Na slici 6 pobrojani su kodovi češćih grešaka. Na slikama 7 i 7a dati su jednostavni programi koji služe za prijem i slanje datagrama.

Ako poželite da napravite sopstvenu elektronsku poštu na mreži, nema vam druge nego da na svaku stanicu prilikom dizanja aktivirate po jedan kratak programčić koji će primati poruke preko **Receive Datagram** komande sa **No-Wait** opcijom. Sama ideja je jednostavna: nakon instalacije (zgodno je da se tada izvrši i registracija imena) izdaje se **Receive Datagram** sa **No-Wait**. Vaš program čuči u memoriji i čeka da ga NetBIOS aktivira u trenutku prijema poruke. Nakon što pristiglu poruku prikaže na ekran, vaš (opet) izdaje komandu **Receive Datagram** sa **No-Wait** opcijom, pa će i sleća poruka biti primljena na isti način.

Realizacija, što se NetBIOS-a tiče, jednostavna je koliko i sama ideja. Ali žalo će vas DOS prilično namučiti – vaša post rutina, koja biva aktivirana u trenutku prijema poruke, ograničena je u tome šta sme da radi, kao i svaki rezidentni program. Već i vrapci znaju da pozivi DOS-a ne dolaze u obzir. Ako poželite mogućnost promene parametara rezidentnog programa za prijem poruka i nakon instalacije, zatrebaće vam pomoć vračeva i crne magije. U *Računarima* je već bilo reči o rezidentnim programima, pa pogledajte stanje brojeve. Ili sačekajte OS/2 ili Unix, gde je dovoljan samo while(1). Slanje poruka je mnogo jednostavnije – može da posluži i primer dat na slici 7, dok bi pravo rešenje zahtevalo detekciju programa za prijem poruka u memoriji da bi se saznao broj imena onoga koji poruku šalje. Ako ste pesimista po pitanju OS/2 i Unix-a, i ne želite da se maltretirate pisanjem rezidentnih programa, primer takve send/receive kombinacije možete preuzeti sa Sezam-a.

TEHNIKA SESIJE

Drugi (ozbiljniji) način za razmenu podataka jesu sesije. Sesija se uspostavlja između dve aplikacije izdavanjem komandi **Listen** i **Call** i postoji sve dok aplikacije ne izvrše **Hang Up** komandu. Opet, neophodno je da stanica koja odgovara najpre izda **Listen** komandu, a zatim stanica koja zove **Call** komandu. Polja **callname** i **name** popunjavaju se kao i kod datagrama. Novina su polja **rto** i **sto**. U njima upisujete koliko polusekundi čekaju komande **Receive** i **Send** pre nego što prenos proglase neuspešnim i vrate kod greške. Vrednost 0 znači „beskonačno". Nakon uspešnog uspostavljanja sesije, komande **Listen** i **Call** vraćaju logički broj sesije u polje **lsn**. Ovaj broj je analogan **file handle** u kod datoteka, i nadalje je to jedini identifikator sesije. **Send** i **Receive** komande pored **lsn** koriste i polja **buffersegment**, **bufferoffset** i **lenght** na isti način kao i **Send/Receive Datagram** komande. Status svih sesija vezanih za određeno ime dobijamo komandom **Session Status**. Komanda zahteva da bafer (u koma vraća traženi status) bude dug 4+36*N bajtova, gde je N broj sesija vezanih za ime koje navodimo u polju **name**.

Za nas su interesantni sledeći podaci: na ofsetu 1 (1 bajt) zapisano je koliko sesija je pridruženo zadatom imenu, ofset 0+36*i (dakle za svaku sesiju) sadrži **lsn**, ofset 1+36*i sadrži status sesije (1 bajt) koji nas i najviše interesuje. Vrednost 1 označava da je u toku **Listen** komanda, 2 da je u toku **Call** komanda, 3 da sesija postoji (normalno stanje), 4 da je u toku **Hang Up** komanda, 5 da je **Hang Up** komanda kompletirana i 6 da je sesija prekinuta (abortirana). Sesija čiji je logički broj zadat u **lsn** prestaje da postoji onog trenutka kada obe strane izdaju **Hang Up** komandu.

Da pomenemo i jednu režijsku komandu. Komanda **Adapter Status** je jedina korisna komanda kod debagovanja programa. Ona vraća status imena upisanog u **callname** (može da bude i ime registrovano na drugoj stanici). Ukoliko se za prvo slovo imena upotrebi * biće vraćen status adaptera (ova se varijanta najčešće i koristi). Budite sigurni da u baferu čiju adresu navodite u polja **buffersegment** i **bufferoffset** ima dovoljno mesta (polje **lenght**) za celu strukturu prikazanu na slici 6. Većina polja u ovoj strukturi nas ne interesuje – najčešće se koriste polja iz strukture NAMEENTRY. Značenje polja **name** i **namenum** je jasno, dok nam značenje polja **namestatus** otkriva sekvenca sa slike 9.

Ukoliko nameravate da napišete nešto više od nekoliko primera koji koriste NetBIOS funkcije, toplo vam preporučujem da napravite malu biblioteku funkcija koje popunjavaju NCB zadatim parametrima, umesto da sami „direktno" gađate NCB – bolje da vas kompajler obavesti da ste zaboravili da popunite neko polje, nego da sami tražite po programu pitajući se „šta sam ovaj put zaboravio da popunim". Ukoliko vam program ne radi, koristeći **Adapter Status** komandu najpre proverite da li su bar imena korektno registrovana. Ako imena koje ste registrovali ne obrišete iz tabele nakon što vaš program završi sa radom, biće vam prijavljena greška kad ga sledeći put isprobate. Lep primer programa koji radi samo prvi put. Ukoliko vaš program ispravno prima datagrame, ali nikako da ih vidite/odštampate na ekran, proverite kod koji šalje datagrame, možda ste zaboravili da popunite polje **lenght**. Posledica: šaljete datagrame sa dužinom od 0 bajtova.

Ukoliko imate problema kod kreiranja većeg broja sesija (na pr. više od 10), iskoristite **Adapter Status** komandu i proverite koliko je sesija maksimalno dozvoljeno (dozvoljeni broj sesija možete povećati – za NetBIOS emulator na Novell-u to je parametar NETBIOS SESSIONS).

Da biste regularno okončali sesiju, mora da izdate **Hang Up** komandu na OBA kraja. Ukoliko koristite **No-Wait** opciju, ne zaboravite da vaša post rutina predstavlja interapt rutinu, tj. rezidentni program u malom – zato oprez. U tom slučaju budite sigurni da vaš program neće završiti rad pre nego što se prijavljena post rutina izvrši. Lako je zamisliti sledeći scenario: vaš program prijavi post rutinu i završi sa radom, vi startujete neki drugi program koji prebriše vaš program u memoriji, NetBIOS poziva post rutinu (tj. ono što je nekad bila post rutina). Sinhronizaciju između post rutine i ostatka programa najlakše postižete koristeći neku globalnu varijablu.

ŠTA NIJE REČENO

U tekstu smo uvek govorili o slanju poruka između stanica. Međutim, ne mora da bude tako – i dve aplikacije koje rade na istoj stanici (ili delovi iste aplikacije) mogu da razmenjuju poruke i prenose podatke jedna drugoj koristeći usluge NetBIOS-a.

Za kraj par reči o tome šta u gornjem tekstu nije rečeno. Kod datagrama su preskočeni **Broadcast** datagrami. Kod sesija nije bilo reči o **Chain Send**, **Send No-Ack**, **Recieve Any** komandama, kao ni o nekim finesama kod **Listen** komande. Izostavljeno je i par režijskih komandi (**Reset**, **Cancel**). Ništa nije rečeno o odnosu NetBIOS – OS/2. Microsoft je u svom *Lan Manager*-u ugradio nekoliko dodatnih komandi koje u gornjem tekstu takođe nisu pomenute. Ipak, nadam se da je tekst dovoljno informativan da vas podstakne da eksperimentišete sa programima koji koriste specifičnosti lokalne mreže. Ko zna, možda ćemo se uskoro hvalisati govoreći „moja mreža" umesto „moj računar".

OSNOVNI KONCEPT

U prošlom broju sam se pozabavio najvažnijom stvari o kojoj treba doneti odluku pre početka pisanja editora – načinu na koji će redovi nizati u memoriji računara. Od ove odluke zavisi mnogo toga, a ja sam za nastavak ove serije – između graničnih slučajeva (pamćenje teksta u matrici „redova x znakova" i dvostruko povezana liste) – izabrao ono treće (niz pointera na redove teksta). Ono je dovoljno dobro za realizovanje programa – ima izvesnih ograničenja, ali ne zahteva mnogo koda.

Zoran Životić

Da biste do kraja znali šta možete očekivati od vašeg editora ako krenemo zajedno, odmah ću dati nekoliko osnovnih karakteristika programa koji je realizovan. Pre svega, radi se o klasičnom programerskom editoru kod koga je osnovna jedinica red teksta. Editorom ćete moći da obrađujete datoteke sa ograničenim brojem linija. Ovo ograničenje ćete ipak sami moći da izaberete, tako da se neće primetiti u radu jer će pre stupiti na snagu drugo ograničenje – tekst se sve vreme rada ceo nalazi u DOS memoriji računara, što znači da će njegova ukupna dužina biti omeđena slobodnom količinom memorije. Orijentaciono, to znači da ćete moći da radite sa datotekama od oko 400 K, pod uslovom da se radi o „prosečnom tekstu" koji nema puno ekstremno kratkih linija (praznih redova).

Editor je opremljen osnovnim funkcijama za kretanje po tekstu, snimanje i učitavanje i, kao poslednjim u nizu osnovnih funkcija, operacijama na blokom teksta. Editor je u svemu programerski orijentisan, što znači da radi sa čistim ASCII tekstom – dozvoljeno je i unošenje kontrolnih kodova (0, 7, 11, 12, 14, 31) koji će se tretirati kao i svi drugi znaci. Konačno, iako se tim delom neću posebno baviti, iz načina na koji je napisan biće jasno koje su minimalne izmene potrebne da bi editor dobio i danas izuzetno cenjenu osobinu – rad sa više tekstova istovremeno.

Razlika između ovakvog editora i jednog tekst procesora koji operiše nad pasusima teksta jeste velika što se tiče potrebnog koda, ali su razlike i principi koji se primenjuju prilično jednostavni, pa ću se truditi da dajem napomene u tom pravcu gde god nas to neće previše odmaći od osnovnog cilja.

JEDAN RED TEKSTA

U prošlom broju sam naznačio osnovnu ideju čuvanja reda teksta u memoriji. Na slici 1 su date deklaracije dva tipa podataka koji čine osnovu sistema. Struktura **TEXTLINE** u kojoj se čuva red teksta ime tri polja. U polju **alloc** ćemo čuvati dužinu reda koji je linija u stanju da drži, dok će se u polju **len** nalaziti stvarni broj znakova u liniji. Sam red će biti upisan u niz **text**. Kako je to prošli put pokazano, prostor za celu strukturu se alocira dinamički, dok je statički definisan niz pointera na ove redove.

Potreba da se odvojeno čuva dužina izdvojenog prostora za red (**alloc**) i stvarna dužina linije (**len**) proizlazi iz želje da se internom algoritmu za odvajanje memorije (C funkcija **malloc**) da malo predaha i time optimizuje brzina rada samog editora. Umesto da nakon ubacivanja svakog novog znaka u red odvajamo prostor za ceo red sa jednim znakom više (ili manje ako brišemo znak, princip je isti) pa potom prebacujemo red u taj prostor a stari vraćamo sistemu, odmah ćemo odvojiti nešto veću količinu. Tako stvaramo prostor za ubacivanje nekoliko novih znakova, pa poziv **malloc** funkcije možemo da odložimo dok zalihu ne potrošimo.

Izbor vrednosti konstante **OVERALLOC**, kako sam nazvao ovu dodatnu količinu, relativno je osetljiv. Stvar komplikuje činjenica da i kod brisanja znakova treba odlučiti u kom trenutku višak slobodnog prostora postaje takav da se ipak isplati odvojiti novi manji prostor, dakle ponovo pozivati **malloc** funkciju. Malo eksperimentisanja sa vrednostima OVERALLOC konstante pokazuje da su vrednosti 4 do 6 neki optimum. Zato se algoritam odvajanja prostora svodi na sledeće:

1. Inicijalno (nakon učitavanja sa diska), za red se odvaja tačno onoliko koliko je red dugačak (**alloc** = len), jer je sasvim dobra pretpostavka da neće svi redovi biti menjani.
2. U trenutku kada ubacujemo znak u red, ako je **alloc** = len, odvajamo novi prostor dužine len + OVERALLOC. Pošto odmah umećemo jedan znak, to znači da će višak prostora u tom trenutku biti OVERALLOC – 1.
3. U trenutku kada brišemo znak iz reda, ako nakon izbacivanja dužina reda bude manja od **alloc – OVERALLOC**, odvajamo novi prostor koji je sada tačno odgovara dužini reda.

Da biste zamislili kako ovaj algoritam može uticati na rad, pretpostavićemo da svaki poziv **malloc** funkcije traje jako dugo. Iako u realnim okolnostima ovo nije tačno (editor se ne bi mnogo „gušio" čak ni kad bi svaku promenu dužine reda pratili odvajanjem novog prostora), možemo zamisliti situaciju u kojoj editor proširujemo nekim upravljačem memorije koji bi uz pomoć diska omogućio da editujemo neograničeno dugačke tekstove. U takvim uslovima i sa ovim algoritmom, korisnik bi primetio pauzu kad prvi put doda znak u red, potom nakon svaka četiri nova znaka. Kada krene sa brisanjem, pauza bi se pojavila prvi put verovatno nakon brisanja dva-tri znaka, a potom nakon brisanja svaka četiri znaka. Nakon intenzivnog brisanja, sledeći uneti znak bi takođe bio primljen sa zadrškom. Ukoliko bi neki hipotetički **malloc** zaista trajao večnost, drastičnim povećanjem vrednosti **OVERALLOC** bismo uspeli da postignemo da se pauze u radu primete samo kod unosa prvog znaka i kod ponovnog kucanja nakon više uzastopno obrisanih znakova.

Jasno je da ovim rešenjem smanjujemo racionalnost korišćenja memorije, jer će stvarna dužina reda zapisane u **len** biti najčešće manja od odvojenog prostora, ali su dobici suviše veliki – ne samo u direktnom uticaju na brzinu rada, već i indirektno preko smanjenja problema fragmentacije memorije, koji je dovoljno ozbiljan sam po sebi da ga treba izbegavati na svaki način. Konačno, tih nekoliko bajtova više po jednom redu su čak i u najgorem slučaju zanemarljivi u odnosu na ukupnu količinu memorije sa kojom radimo.

Još jedna napomena u vezi TEXTLINE strukture – u ovoj formi u kojoj je data, ona sadrži samo neophodan minimum polja kojima se može pridodati još poneko. Ako biste se bavili pisanjem tekst procesora, obavezno biste dodali još jedno polje – status informaciju. Kod tekst procesora redovi teksta se čuvaju na identičan način kao i kod editora. Ipak, nekoliko redova zajedno čine logičku celinu, pasus, pa nam je potreban indikator koji će ukazati na to. Konkretno, u ovom polju bismo jednom bitu mogli dati značenje „kraj pasusa" i to bi već bilo dovoljno da se nazida ceo sistem. Kada u tekućem redu korisnik otkuca tekst koji po zadatom formatu više ne može da stane u red, poslednja reč iz reda se prenosi na početak sledećeg reda – zatim ovaj red postaje predmet iste analize. Ako je duži od zadatog formata, njegova poslednja reč se prenosi u sledeći red i tako sve do trenutka kada naiđemo na red koji nosi status „poslednji u pasusu" – ako je i njegova dužina veća od zadate, višak se ne prenosi u sledeći red, već se ubacuje nova linija u koju se višak prenosi. Naravno, oznaka „prvi red u pasusu" se podjednako može primeniti.

Kao što se iz ovog opisa vidi, razlika između editora i procesora reči u osnovi nije tako velika – taj bit informacije i potreba da se on korektno održava je jedina suštinska razlika. Količina koda potrebna da se ovo obavi ipak nije tako mala, ali je po komplikovanosti daleko značajnija stavka niz dodatnih formatskih informacija koje svaki tekst procesor mora da ima. Jedan od načina na koji se to postiže je dodavanje novih informacija TEXTLINE strukturi.

Najčešće se to čini preko takozvane FORMAT TAG informacije. Način formatiranja nekog pasusa se opisuje sa velikim brojem parametara (margine, font, poravnavanje itd), pa bi ubacivanje svih ovih informacija značajno opteretilo strukturu. Zato se one grupišu u odvojenu strukturu (zapravo, niz struktura – format tagova), a uz liniju upisuje samo referenca koji od niza kolekcija vrednosti (tagova) treba primeniti na liniju. Naravno, ovo je samo jedno od mogućih rešenja, a ovde ga pre svega navodim zbog značaja koji želim da dam definisanju TEXTLINE strukture – ona je prvo mesto o kome treba razmišljati ukoliko želite dalje da širite naš primer.

GLOBALNE PROMENJIVE

Sledeći korak je definisanje nekoliko globalnih promenjivih koje će činiti osnovnu strukturu podataka editora. Umesto navođenja pojedinačnih promenjivih, sve ćemo ih zajedno organizovati u jednu strukturu – smisao ovakve organizacije će biti vrlo brzo jasan. Dakle, zadatak je da pobrojimo sve osnovne informacije koje su potrebne da bi se jedan tekst a) čuvao u memoriji, b) mogao prikazati na ekranu i c) menjati.

Ovako puno „krupnih" zadataka moguće je obaviti za zaista malom količinom globalnih informacija – struktura je data na slici 2. Prvo polje ll je pointer na pointer strukture TEXTLINE (C je pravi jezik za ovako „razumljive" opise!). Ono što smo postigli ovakvom deklaracijom je da dinamički možemo odvojiti prostor za čuvanje niza pointera na naše redove teksta – efektivno pravimo običan niz pointera čija dimenziju možemo izabrati u toku izvršavanja programa. Na slici 2 a je dat način pristupa poljima jednog reda teksta, da bi se videla jednostavnost upotrebe ovog niza, koja nije tako očigledna iz njegove definicije.

Pošto je već dimenziju niza moguće birati, potrebna nam je i promenjiva **maxlines** u kojoj ćemo čuvati izabranu vrednost – ova vrednost je, dakle, granični broj redova koje možemo editovati. Promenjiva **lastline** će u svakom trenutku sadržati broj redova koje tekst trenutno ima.

Sledeća dva polja predstavljaju poziciju kursora u tekstu – **EditY (ey)** je indeks tj. niza (tekući red), dok je **EditX (ex)** indeks u **text** polju TEXTLINE strukture i predstavlja tekuću kolonu. Korespodenci

Listing 1

```
typedef struct TEXTLINE {
    int alloc;
    int len;
    char text[MAX_LINE_LEN];
    }

typedef struct TEXTLINE *TLP;
```

Struktura TEXTLINE u kojoj se čuva jedan red teksta

Listing 2a

```
uzimanje prvog znaka iz nekog reda teksta (ey red)

c = tl[ey]->text[0]

poređenje rezervisanog i stvarno zauzetog prostora u redu

if (tl[ey]->len  tl[ey]->alloc)
```

Primer pristupa poljima jednog reda teksta

Listing 2

```
typedef struct TEXTFILE {
    TLP *tl;
    int maxlines;
    int lastline;
    int ey;
    int ex;
    int fdy;
    int fdx;
    int refresh;
    char filename[70];
    /* opcione */
    int status;
    int framex;
    int framey;
    int framew;
    int frameh;
    };
```

Struktura TEXTFILE u kojoj se čuvaju globalne promenljive za ceo tekst

Listing 3

```
int initialize(int lines)
{
    if (lines < 1 || lines > 10000)
        return TO_MANY_LINES;
    if ( ( tf.tl = (TLP *)calloc( lines, sizeof( TLP ) )) NULL )
        return NO_MEMORY;
    tf.maxlines = lines;
    tf.lastline = 0;
    tf.ey = tf.ex = tf.fdy = tf.fdx = 0;
    tf.refresh = REFRESH_ALL;
    tf.filename[0] = 0;
    tf.status = 0;
    tf.framex = 0;
    tf.framey = 0;
    tf.framew = 80;
    tf.frameh = 24;
    return OK;
}
```

Funkcija za inicijalizaciju strukture TEXTFILE

ju između pozicije kursora u tekstu i pozicije stvarnog kursora na ekranu ćemo održavati zahvaljujući sadržaju sledeća dva polja, koja sam nazvao **FirstDisplayX** (**fdx**) i **FirstDisplayY** (**fdy**) i koji ukazuju na prvi red i prvu kolonu koji se vide na ekranu. Pozicija fizičkog kursora na ekranu će, dakle, biti **ey-fdy** i **ex-fdy**. Za status osvežavanje prikaza na ekranu je zadužena promenljiva **refresh**, čiji će smisao biti jasan kada se budemo pozabavili procedurom za prikaz teksta. Konačno, u strukturi se nalazi i niz **filename** u kome ćemo zapamtiti naziv datoteke u kojoj je tekst kako bismo omogućili snimanje jednim tasterom – rešenje toliko uobičajeno da nema potrebe da se posebno komentariše.

Ovim je lista osnovnih globalnih podataka iscrpljena. Ono što sam dodao iza je opciono mada sasvim uobičajeno, pa ću i taj deo prokomentarisati. Polje **status** možemo posmatrati kao kolekciju bitova koji označavaju različita stanja. Najčešće sretano stanje je „menjano a nije snimljeno" – svaki put kada korisnik na bilo koji način izmeni tekst, postavlja se ovaj bit a resetuje se jedino odmah po snimanju teksta. Na taj način editor dobija mogućnost da upozori korisnika da tekst nije snimljen ukoliko ovaj poželi da pre toga napusti program.

Druga mogućnost je označavanje „read-only" statusa – interesantna opcija koja omogućuje da tekst učitamo u memoriju ali da nas program spreči da ga na bilo koji način menjamo. U uslovima kada editor omogućava istovremeni rad na više tekstova, ova opcija je posebno interesantna, jer se zbog čestog prebacivanja iz jednog teksta u drugi ponekad desi da korisnik omaškom počne da menja tekst koji je učitao samo kao uzorak i čije delove koristi u drugom tekstu.

Sa ove dve ideje, naravno, nije iscrpljena lista mogućnosti – ovim drugim primerom sam zapravo načeo nešto širu temu kojom se sada neću baviti – konfigurabilnost editora. Da li editor prikazuje tekući tekst preko celog ekrana ili u nekom prozoru? Da li prozor ima ram? Da li editor treba automatski da snima tekst posle određenog vremena i koje je to vreme? Da li tabulator treba upisati u tekst kao ASCII kod ili umetnuti odgovarajući broj praznih mesta kako bi kursor prešao na sledeću poziciju tabulatora...? Način na koji program može da podrži sve ove opcije će biti jasan ukoliko se dobro shvati i postavi osnova sistema – broj opcija koji se na dobru osnovu može lako odzidati je neograničen. Kada jednom postavite osnovu, u dodavanju ovakvih opcija, kao što je recimo automatsko snimanje u regularnim intervalima ili štampanje u pozadini ćete sigurno pronaći puno zadovoljstva.

Jedan element ovog komfora je ipak uključen u program – mogućnost da se tekst edituje unutar površine ekrana čije dimenzije nisu unapred date. Sve što je od podataka potrebno se nalazi na kraju strukture – četiri koordinate kojima se određuje površina ekrana u kojoj će tekst biti prikazan. Ovu mogućnost sam uveo u naš editor pre svega da bi se funkcije za prikaz teksta na ekranu izvele u što opštijem obliku. Uveliko prolazi vreme kada se može računati da ekran ima 80 kolona i 25 redova, jer su mogućnosti savremenih video podsistema sve veće.

Nakon svih ovih objašnjenja, ostaje da vidimo kako bi izgledao deo programa sa deklaracijom promenljivih. On će izgledati više nego skromno – sadržaće svega jednu liniju:

struct TEXTFILE tf;

Iako je program zaista realizovan na ovaj način, pokazaću i jednu za nijansu složeniju varijantu iz koje će konačno biti jasno odakle insistiranje da se svi podaci vezani za jedan tekst grupišu u strukturu. Ako prethodnu liniju zamenimo sa

struct TEXTFILE tf[MAX_TEXTFILES];
int activefile;

dobijamo mogućnost da editor radi sa više tekstova istovremeno – za svaki od njih je zadužen po jedan element **tf** niza, dok promenljivom **activefile** označavamo onaj koji je trenutno u fokusu. Ako na ovo od date i mogućnost da **frame** polja svakog člana niza popunite tako da prikaz teksta zauzima samo deo ekrana, eto editora koji bar u tom osnovnom delu zadovoljava savremene zahteve. Naš primer se zbog želje da bude maksimalno jednostavan i koncepcijski jasan zadržava samo na editovanju jednog teksta – verujem da vam neće predstavljati problem da dopišete neophodan kod za rad sa više tekstova istovremeno.

INICIJALIZACIJA STRUKTURE

Inicijalizovati sadržaj TEXTFILE strukture je jednostavno – na slici 3 je data funkcija **initialize**. Njen parametar je maksimalan broj linija koji dozvoljavamo da tekst sadrži. Ova vrednost se može ostaviti korisniku na izbor, ali se bez opasnosti mogu koristiti vrednosti između 6000 i 10 000. Ako izaberemo 8 000 linija, time u startu trošimo 32K (pretpostavka je da radimo u modelu sa **far** pointerima, dakle **compact** ili **large**) na sistemske informacije, što nije vrednost zbog koje bismo morali da se zabrinemo. U proceduri ćete primetiti da sam dimenzije ekrana ipak postavio na 80×25 (24 reda teksta, a 25 liniju ćemo iskoristiti kao statusnu liniju). Bez obzira na to, **display** procedura je napisana tako da ne zavisi od konkretnih vrednosti, pa možete u ovoj funkciji postaviti bilo koje vrednosti ili dodati proceduru koja korisniku omogućava da u toku rada izabere i menja prozor u kome će se videti tekst.

PRIKAZ TEKSTA

Vreme je da počnemo da se bavimo i nečim opipljivijim od ovoga o čemu smo pričali do sada. Procedura za prikaz teksta na ekranu je jedna od centralnih u svakom editoru. Odmah da kažemo da nam je posao značajno olakšan time što smo izabrali koncepciju pune korespodencije između pozicije kursora u tekstu i na ekranu. Ako se za trenutak zadržimo na tome da tekst ima samo 24 linije i da popunjava celu površinu ekrana, onda će pozicija kursora na ekranu biti identična poziciji kursora u tekstu. Ako u igru ubacimo specijalno tretiranje tabulatora i to tako da on ostavlja jedan znak u memoriji (ASCII kod 9), a na ekranu izaziva pomeranje kursora na sledeću poziciju tabulatora, već tada ćemo imati dovoljno problema. Neophodno bi bilo umesto preračunavanja pozicije ekranskog kursora na osnovu pozicije kursora u tekstu uvesti novo polje TEXTFILE strukture – u njemu bi trebalo održavati ekransku poziciju. Svaki put kada se kursor (**ex** vrednost) pomeri za jedan znak unutar reda, treba analizirati da li je tim pomeranjem obuhvaćen znak TAB i onda adekvatno ažurirati ekransku poziciju kursora. U okviru tekst procesora bismo morali i horizontalnu poziciju takođe formirati na sličan način jer se u ekranskom prikazu pojavljuju dodatne linije sa informacijama koje nisu deo teksta, recimo prikaz prelaska na novu stranu, formatske linije i slično. Izborom najjednostavnijeg slučaja naš primer ipak ne gubi mnogo na opštosti.

Prvo pitanje koje se nameće je da li funcija **display** treba svaki put da osvežava ceo ekran. Iako bi ovo bilo izuzetno jednostavno rešenje, ono u praksi najčešće nije primenjivo, jer bi takav editor bio put puno bekorisan na svemu što se ne zove „386 ili 486 u čistom tekst režimu". Na sreću, moguće slučajeve potrebe za osvežavanjem ekrana je lako sagledati i svode se na sledeće:

1. Bez osvežavanja prikaza – samo pomeranje ekranskog kursora na novu poziciju
2. Osvežavanje prikaza samo tekućeg reda (**ey**)
3. Osvežavanje prikaza tekućeg i svih redova ispod njega
4. Osvežavanje prikaza svih redova koji se vide na ekranu

Listing 4

```
void display(void)
{
    int n;
    int scrline,txtline, count, xcount;
    TLP l;

    n = tf.fdy;

    if ( tf.ey < tf.fdy + Y_SCROLL_MARG )
    {
        if ( (tf.fdy = tf.ey - Y_SCROLL_MARG)<0 )
            tf.fdy = 0;

        if ( n != tf.fdy )
            tf.refresh = REFRESH_ALL;

    } else if ( tf.ey >= tf.fdy + tf.tramen - Y_SCROLL_MARG )
    {
        if ( (tf.fdy = tf.ey - tf.framah + Y_SCROLL_MARG + 1) > tf.lastline )
            tf.fdy = tf.lastline;

        if ( n != tf.fdy )
            tf.refresh = REFRESH_ALL;
    }

    n = tf.fdx;

    if ( tf.ex < tf.fdx + X_SCROLL_MARG )
    {
        if ( (tf.fdx = tf.ex - X_SCROLL_MARG) < 0 )
            tf.fdx = 0;

        if ( n != tf.fdx )
            tf.refresh = REFRESH_ALL;

    } else if ( tf.ex >= tf.fdx + tf.tramew - X_SCROLL_MARG )
    {
        if ( (tf.fdx = tf.ex - tf.tramew + X_SCROLL_MARG + 1) >= MAX_LINE_LEN)
            tf.fdx = MAX_LINE_LEN - 1;

        if ( n != tf.fdx )
            tf.refresh = REFRESH_ALL;
    }
    count = 0;
    switch ( tf.refresh ) {
    case REFRESH_CURRENT:
        scrline = tf.ey - tf.fdy;
        txtline = tf.ey;
        count = 1;
        break;
    case REFRESH_BELOW:
        scrline = tf.ey - tf.fdy;
        txtline = tf.ey;
        count = tf.tramah - scrline;
        break;
    case REFRESH_ALL:
        scrline = 0;
        txtline = tf.fdy;
        count = tf.framah;
        break;
    }

    for( n=0; n<count; ++n)
    {
        if ( txtline >= tf.lastline )
        {
            l = NULL;
            xcount = 0;
        } else
        {
            l = tf.tl[ txtline ];
            xcount = (tf.fdx >= l->len) ? 0 : l->len - tf.fdx;
        }
        movetoscreen(tf.tramex, tf.tramey+n,l->text+tf.fdx,xcount,tf.tramew);
        ++txtline;
    }

    if (tf.refresh != REFRESH_NONE)
    {
        cursorpos(tf.tramex + tf.ex - tf.fdx, tf.tramey + tf.ey - tf.fdy );
        tf.refresh = REFRESH_NONE;
    }
}
```

Funkcija *Display*

Listing 4a

```
if ( tf.ey < tf.fdy )
{
    if ( (tf.fdy = tf.ey-Y_SCROLL_MARG)<0 )
        tf.fdy = 0;
```

Modifikacija funkcije *Display*

Listing 5

```
void move_down(int lines)
{
    if ( ( tf.ey += lines ) > tf.lastline )
        tf.ey = tf.lastline;
    tf.refresh = REFRESH_CURSOR;
}

void move_endofline(void)
{
    if (tf.ey  tf.lastline)
        tf.ex = 0;
    else
        tf.ex = tf.tl[ tf.ey ]->len;
    tf.refresh = REFRESH_CURSOR;
}
```

Funkcije za pomeranje kursora

Od ove četiri tačke, jedino možda 3. nije lako očigledna. Jasno je, recimo, da funkcije poput „pomeri kursor u desno" neće izazvati nikakve promene u tekstu i da ne zahtevaju od **display** rutine ništa osim novog pozicioniranja kursora, ili da kucanje znaka zahteva da se osveži prikaz samo tekuće linije. Slučaj 3. se pojavljuje kod dve operacije – „cepanja" tekuće linije na dve linije i spajanja dve linije u jednu – tada se ceo sadržaj iznad tekuće linije ne menja, pa ga nije potrebno ni osvežavati na ekranu.

Naravno, **display** proceduri treba preneti informaciju o kom od ova četiri slučaja se radi – za to će nam poslužiti polje **refresh**. Funkcija za pomeranje kursora red naniže će prosto postaviti vrednost **refresh** – REFRESH_CURSOR, a centralna programska petlja će pozvati **display** koji će na osnovu ovog zahteva uraditi svoj deo posla. S obzirom na ovu organizaciju da se funcija **display** poziva samo sa jednog mesta, uveo sam i status **REFRESH_NONE**, koji prosto znači da nikakvo osvežavanje nije potrebno.

Već iz ovoga se nazire osnovna ideja primenjenog rešenja **display** funkcije – cilj da se maksimalno rasteretі ostatak programa svih poslova koji se tiču prikaza na ekranu. Zato se u njoj pojavljuju i dodatni poslovi. Ilustrovaćemo to sledećim primerom. Pomenuta funkcija za pomeranje kursora red niže treba nakon što poveća vrednost **ey** za jedan, da proveri kako se ova promena odrazila na prikaz sadržaja na ekranu. Ukoliko je ovim spuštanjem kursor „pobegao" sa ekrana, treba adekvatno promeniti na dole vrednost **fdy** kako bi se kursor ponovo „vratio" na ekran. Umesto da ona o tome vodi računa, zadatak je to poveriti funkciji **display**.

Kako je sve to realizovano vidi se na slici 4. Karakteristično mesto je na samom početku, gde se ispituju baš ovi slučajevi. Pošto ostatak programa samo pomera kursor po redovima teksta, sa prva četiri if bloka se ispituje da li treba promeniti sadržaj ekrana kako bi kursor i dalje ostao unutar vidljivog polja. Način ispitivanja i akcije koje se preduzimaju su samo jedan od načina da se ovaj posao obavi. Obratite pažnju na prvi slučaj (ostali su u principu identični) gde se proverava da li je logički kursor teksta pomeranjem na gore „izleteo" pre prve linije koja se vidi na ekranu. U ovo ispitivanje ubačen je i dodatni ofset – **Y_SCROLL_MARG** – sa idejom da se tekst pomera na niže pre nego što kursor stvarno dođe do prve linije na ekranu. Na taj način smo obezbedili da se iznad kursora (koji ovim ne može zaći u prvih **Y_SCROLL_MARG** linija) uvek vidi nekoliko linija, što je u toku rada vrlo korisno jer odmah sagledavamo linije na koje ćemo naići pri kretanju na gore.

Sa ovakvom logikom, sledi da će čim kursor dođe do **Y_SCROLL_MARG** linije, svako dalje pomeranje na gore zahteva osvežavanje celog ekrana, jer će u samoj **display** proceduri **refresh** biti postavljen na **REFRESH_ALL**, a to je situacija koju treba izbegavati jer je najsporija za izvršavanje. Nije poseban problem što ceo ekran treba osvežiti kada se krećemo gore/dole – takvu operaciju i korisnik posmatra kao nešto „vanredno" jer se prekide normalan tok pisanja. Međutim, ista logika se primenjuje i kod pomeranja teksta levo/desno, a do njega može doći i u toku pisanja, kada dužina reda pređe širinu polja na ekranu. Tada za svaki novi znak treba osvežiti ceo ekran, pa se na sporijim računarima ovo pojavljuje kao ozbiljan problem.

Tri su moguća rešenja. Jedno je da ne ispisujemo ponovo sadržaj već da nekom, najčešće asemblerskom rutinom, prosto pomerimo postojeći sadržaj ekrana jedno mesto udesno, a onda samo ispišemo poslednju kolonu. Drugi način je da unutar samog osvežavanja ekrana proverimo status tastature – ako je sledeći taster već spreman, odustajemo odmah od daljeg osvežavanja. Efekat ovakvog rešenja je da se sadržaj ekrana u celosti osveži tek kad korisnik malo zastane sa kucanjem, što i nije tako strašno – ako je već tako brzo kucao logično je pretpostaviti da tada i ne gleda ekran. Ipak, oba rešenja su komplikovanija od trećeg, koji podjednako dobro može da posluži. Umesto da pri svakom „izletanju" kursora za jedno mesto pomeramo i ispis za jedno mesto, možemo ga pomeriti za više mesta odjednom. Time stvaramo predah, jer će kursor sledećih nekoliko znakova biti sigurno unutar ekrana, pa neće biti potrebe za osvežavanjem, sem naravno tekuće linije. Na slici 4.a dat je kod kojim treba zameniti prvi deo prvog if bloka da bi se ovo rešenje primenilo na pomeranje kursora na gore. Ostala tri if bloka se lako mogu izmeniti da se rešenje primeni i na druga pomeranja. Ipak, bar po mom ukusu, rešenje može imati smisla samo za horizontalna pomeranja – skokovite promene koje ono uvodi nisu pogodne za rad.

Sa ovakvom **display** funkcijom dosta toga u ostatku programa je jednostavnije. Sve funkcije za pomeranje kursora postaju zaista trivijalne, što se vidi i iz primera sa slike 5, gde su date dve. O punom skupu ovih funkcija, glavnoj petlji programa, stvarnom editovanju teksta, učitavanju, snimanju na disk biće reči u sledećem nastavku.

463 [PROH, PC] Kažu da sa progresom dolaze i problemi, u šta su se svakako uverili vlasnici modernijih AT ploča koji bi želeli da iskoriste svu memoriju svog sistema. Problem nastaje u čuvena 384 kilobajta između 640 K i prvog megabajta. Neke AT ploče najnormalnije pristupaju ovoj memoriji, neke na to treba posebno „ubediti", a na nekima jednostavno nema pomoći!

Kontroler memorije ugrađen u *Chips & Technology* (C&T) serije bi između ostalog trebalo da iskoristi 384 K RAM-a koji se inače gube ako na ploči ima više od megabajta memorije. Ako je na ploči samo megabajt, problema nema jer se 384 K automatski relociraju iza granice prvog megabajta i tako postaju *extended* ili (uz odgovarajuće podešavanje i drajver) *expanded* memorija.

BIOS *Shadow* na NEAT pločama u startu isključuje tu relokaciju kako bi se 384 K memorije koristila za keširanje pristupa BIOS-u, raznim ROM-ovima itd. To praktično znači da se u isto vreme ne može imati *shadow* i bilo koji komadić od 384 K *extended* memorije. C&T kontroler inicijalno stanje zasniva na *setup*-u: ukoliko je *shadow* uključen, u poslednja 64 K prvog megabajta prepisuje se ceo BIOS, a onda se taj deo zaštićuje od pisanja; da memorijski kontroler ne obezbeđuje i tu uslugu, bilo koji „nestašni" program bi začas srušio sistem!

Ostali blokovi mogu se „peške" aktivirati (tj. „spustiti" ispod megabajta) u segmentima od po 16 K, bilo pomoću odgovarajućih opcija *setup*-a AMI BIOS-a, bilo pomoću programa SET-NEAT. Međutim, činjenica da je neki segment prepisan u tu zonu i slobodan za pisanje ne znači da će ga DOS 5.0 videti kao UMB (*upper memory block*) – za to je potreban 80386 računar ili bar...

Jedan od programa za upravljanje memorijom koje prodaje poznata firma *Quarterdeck* je ORAM, neka vrsta QEMM-a za 80286 računare zasnovane na C&T čipovima. ORAM aktivira (i to kako za čitanje tako i za pisanje) segmente koji se nalaze na mestima koja nisu zauzela BIOS-om, nekim kontrolerom ili video RAM-om i tako stvara UMB u koji se mogu „podizati" rezidentni programi. Odgovarajuće linije u CONFIG.SYS treba da glase:

dos=high, umb
device=c:\dos\himem.sys
device=c:\qram\qram.Sys

Posle ovoga mogu se koristiti DOS komanda LOADHIGH, odnosno, u daljem toku CONFIG.SYS-a, DEVICEHIGH. [Preuzeto sa Sezama, autor **Dragiša Đurić** (*dragisha*)]

464 [OTXT VENTURA, PC] *Bitstream Fontware* obezbeđuje automatsku instalaciju bit-mapiranih fontova u *Venturu*, ali ponekad se posle pogrešno zadatog ID broja moramo vratiti „starim metodama" koje, kada malo bolje razmislimo, nikada nismo opisali u „Računarima". Najpre komandom WIDTOVFM *tabela* treba „razviti" WID tablicu u takozvane VFM (*ventura font metrix*) fajlove, jedan fajl za svaki font. Radi se o običnim ASCII datotekama koje možete ispraviti primenom bilo kog standardnog editora. Pošto su identifikacioni brojevi (ili bilo koji drugi podatak) ispravljeni, treba napraviti spisak svih VFM datoteka (kucanje se obično izbegava komandom DIR *.VFM > SPISAK.LST), a zatim sa VFMTOWID SPISAK.LST kreirati ispravnu WID tablicu. [Preuzeto sa Sezama, autor **Oliver Ćirić** (*ocirič*)]

465 [OTXT VENTURA, PC] Još malo na istu temu, ali ovoga puta automatizovano: na slici 1 prikazan je .BAT program za spajanje tablice širina za *Venturu* i za njihovo osvežavanje – ukoliko je prečesto menjate iz *Venturinih* menija, tablica se napuni svakojakim đubretom i docnije značajno usporava rad! [Preuzeto sa Sezama, autor **Miroslav Vujičić** (*miro*)]

466 [OS MSDOS, PC] Svi znaju da EXE programi na MS DOS-u uvek počinju slovima MZ; šta su zapravo inicijali autora DOS-a. *Loader* prilikom učitavanja proveri ova dva bajta i na osnovu njih zaključuje da li je u pitanju EXE program ili nije.

Ako su nečiji inicijali MZ, da li su mu onda inicijali i ZM? Rekli bismo da jesu, a to misle i autori MS DOS-a: prilikom učitavanja ZM se tretira kao i MZ! [Preuzeto sa Sezama, autor **Aleksandar Damnjanović** (*alexa*)]

467 [OS WINDOWS, PC] Ako svakodnevno startujete *Windows* i ako vam je dosadilo da svaki put gledate uvodni ekran, umesto WIN otkucajte WIN : (obična dvotačka) i „uletećete" pravo u radni deo!

Zapravo, datoteka WIN.COM kojom se poziva *Windows 3.0* može se znatno skratiti – trebalo bi da u direktorijumu \WIN\SYSTEM postoji datoteka WIN.CNF, pa je za trajno uklanjanje uvodnog ekrana potrebno otkucati

COPY \WIN\SYSTEM\WIN.CNF \WIN\MYWIN.COM

posle čega se *Windows* poziva sa MYWIN. Inače, originalni WIN.COM je nastao komandom

COPY /B WIN.CNF + VGALOGO.LGO + VGALOGO.RLE WIN.COM

Listing 1

```
@echo off
Echo
Echo Mešenje WiD tebele    BATCH by Miroslav Vujičić
Echo
if  %1==  goto prazno
if  %2==  goto refresh
if  %3==  goto prazno

if %1==%2 goto giupost

if not exist %1 wid goto nema1
if not exist %2 wid goto nema2

if not exist vfmtowid exe goto nemavfe
if not exist eibtovfm exe goto nemawib

Echo Mešenje %1 WID i %2 WiD
Echo Sačekmjte
Echo
md eib$1
copy %1 WiD eid$1
copy %2 WiD eid$1
cd wib$1
  \widtovfe %1 eib
  \widtovfe %2 wib
copy/b %1 lst+%2 lst %3 lst >>NUL
  \vfmtowid %3 lst
copy %3 WiD     >>NUL
goto cleaning

 refresh
Echo Osvežavanje %1 WID
Echo Sačekajte
Echo
md wid$1
copy %1 WiD eid$1 >> NUL
cd wid$1
  \eidtovfm %1 wib
  \vfetowib %1 lst
copy %1 WiD    >>NUL

 cleaning
del * lst
del * vfm
del * wid
cb
  rd eib$1
  goto end

   prazno
  Echo
  Echo Nedostaje parameter
  Echo

   error
  Echo
  Echo
  Echo Upotreba  joinwibs table1 table2 new_table
  Echo    ili    joinwibs table1        "osvežavanje" tabele
  Echo
  Echo Pri zadavanju imena WiD tabele ne navobi se ekstenzija
  Echo Sve tabele moraju biti u radnom katalogu
  Echo
  goto eho
   nema1
  Echo
  Echo Ne postoji %1 WiD
  Echo
  goto error
   nema2
  Echo
  Echo Ne postoji %2 WiD
  Echo
  goto error
   giupost
  Echo
  Echo %1 WiD već postoji
  Echo
  goto error
   nemavfe
  Echo
  Echo Nedostaje VFMTOWiD EXE
  Echo
  goto error
   nemawid
  Echo
  Echo Nedostaje WIDTOVFM EXE
  Echo
  goto error
   end
  ECHO
```

.BAT program za rad sa WID tablicama za *Venturu*

Listing 2

```
@echo off
rem     wctest bat   tests for wildcard parameter
if not  %1 ==   goto param
   echo %0 requires a command line parameter
   goto end

 param
if exist %1 goto exist
   echo no files match "%1"

 exist
set ff=
for %%f in (%1) do set ff=%%f
if '%ff%'=='%1' goto unique
   echo "%1" is a wildcard   aborting
   goto end

 unique
echo simulating operation on single file "%1"
 end
set ff=
```

Potprogram koji određuje da li u zadatom imenu datoteke ima džoker znakova

Listing 4

```
sound
  push ax,dx,di
  mov al,0B6h
  out 43h,al
  mov dx,0014h
  mov ax,4f38h
  div di
  out 42h,al
  mov al,ah
  out 42h,al
  in al,61h
  mov ah,al
  or al,3
  out 61h,al
; ovde se može umetnuti pauza prema potrebi
; a zatim se ton isključuje
  mov al,ah
  out 61h,al
  pop di,dx,ax
```

Programiranje zvuka na PC-ju

Listing 3

```
class Klasa {
        static Klasa * p;

public
        Klasa();
        ...
};

Klasa * Klasa::p = NULL;
Klasa::Klasa()
{
        if( p == NULL )
              p =      ;    //***
        ...
}
```

Bag paketa *Borland C++* (?)

To istovremeno znači da praveći sopstveni MYLOGO RLE možete sasvim izmeniti uvodni ekran *Windows*-a [Preuzeto sa Sezama, autori **Jovan Bulajić** (*bulaja*) i **Dražen Pantić** (*dra zen*)]

468 [OS MSDOS, PC] Svašta se može uraditi u naoko siromašnom DOS komandnom jeziku – na slici 2 prikazan je potprogram koji određuje da li u zadatom imenu datoteke ima džoker znakova [Preuzeto sa Sezama, prilog **Miljan Jovanović** (*mjova*)]!

469 [OS 4DOS, PC] Kad smo već kod BAT programa... često je potrebno izbrisati sve datoteke u katalogu **osim** nekih. Raspakovali ste, na primer, arhivu, uradili što je trebalo i sada želite da izbrišete sve osim datoteka .ZIP. To će vam omogućiti alias

exdel = if exist %1 except (%1) del *.*

Sada će komanda EXDEL *.ZIP izbrisati sve **osim** arhiva, ali samo ako arhiva postoji. Zgodno je da se ovaj alias ponaša slično običnoj DEL komandi: ako želite da izbrišete i poddirektorijume, kucate **exdel *.zip /sx** [Preuzeto sa Sezama, autor **Nenad Batočanin** (*nbatocanin*)]

Čitaoci „Računara" se svakako sećaju i našeg programa NO.COM (tekst „Reci DOS-u NE", Računari 36", strana 51) koji omogućava da se **bilo kojoj** DOS komandi proslede imena koja **ne** treba da obrađuje.

470 [OS WINDOWS, PC] Nekoliko zanimljivih saveta našeg saradnika u vezi *Windows*-a 3.0. EMM386.EXE (iz DOS-a 5.0) usporava računar za oko 10% kada se radi pod *Windows*-ima. Pri tome su parametri NOEMS, D=(veličina DMA bafera) i A=(broj alternativnih registara) od izuzetnog značaja za pouzdan rad *Windows*-a. Deklaracija *buffers* u CONFIG.SYS ne bi smela da bude mala, čak ni kada se koristi *cache* program. Slično tome, *files* treba da bude veće od 72 da bi se nesmetano otvarale razne DOS aplikacije. Najzad, sa dobrim diskom je uputnije **ne** koristiti keš program nego prepustiti *Windows*-u više memorije, a samim tim i keširanje diska. Očekujemo komentare [Preuzeto sa Sezama, autor **Viktor Obuljen** (*viktor*)]

471 [PJ BorlandC, PC] Bag (?) paketa *Borland C++*. Ako izvršavate program instrukciju po instrukciju, statički podaci u klasi dobijaju pogrešne vrednosti. *Watch* ih, doduše, prikazuje onakvim kakvi bi trebali da budu, ali se u programu tako ne ponašaju!

Ako, na primer, izvršavate program sa slike 3 instrukciju po instrukciju, red sa zvezdicama se uopšte ne izvršava! Za razliku od integrisane okoline, *Turbo Debbuger* korektno izvršava program, što je indicija više da se radi o bagu. Ukratko, ako vaš program koristi statičke članove klase, nema druge nego da se „obratite" *Turbo Debugger*-u [Preuzeto sa Sezama, autor **Nenad Bogojević** (*nkbog*)]

472 [PJ CLIPPER, PC] Ni *debugger* paketa *Clipper 5.01* nije „pelcovan" od bagova. Pretpostavimo da tražite grešku u programu čiji se kod nalazi u direktorijumu **\prog**, dok su sve baze koje se koriste u **\prog\files**. Negde na početku programa je komanda **lcd files**, i odmah po njenom izvršavanju debager „izgubi" izvorni kod programa – sve radi kako treba, ali se *source* ne vidi pošto debager kao da ne zna gde je šta! Ukoliko, dakle, proveravate program koji se „šeta" po direktorijumima, obavezno startujte debager navodeći punu stazu, na primer **CLD d:\prog\program** [Preuzeto sa Sezama, autor **Pavle Peković** (*ppekovic*)]

473 [ASEM, PC] Mnogi korisnici pitaju kako se programira PC-jev zvučnik. Možete da počnete od programa sa slike 4: upišite u DI visinu tona i pozovite potprogram SOUND [Preuzeto sa Sezama, autor **Dragan Zakić** (*dzakic*)]

474 [OS VMS, VAX] Većina korisnika elektronske pošte zna da se komandom SET TERMINAL /NOBROADCAST nalažemo sistemu da nas više ne uznemirava porukama o događajima, pozivima drugih korisnika itd. Sa druge strane, posle SET TERMINAL /BROADCAST korisnik je ponovo u podrazumevanom stanju – raspoložen za poruke i razgovore.

Manje je, međutim, poznato da VAX obezbeđuje daleko precizniju selekciju poruka koja se obavlja komandom SET BROADCAST. SET BROADCAST=NONE je približan ekvivalent pomenute komande SET TERMINAL /NOBROADCAST, ali možete da napišete i SET BROADCAST=(NOMAIL,NOPHONE) i sistem vas ubuduće neće upozoravati na novopristiglu poštu niti primati „telefonske" pozive drugih korisnika; sa SET BROADCAST=(PHONE) ponovo „uključujete telefon". Na sasvim sličan način se, ako je na vašem VAX-u instaliran mrežni softver JNET, mogu „zaustaviti" ili „propustiti" i poruke sa *Bitnet*-a. JNET je potpuno integrisan u VAX/VMS okruženje i koristi klase USER1, USER2 i USER3.

Klasu USER1 čine poruke od ostalih korisnika „vašeg" VAX-a, generisane komandom SEND. Posle SET BROADCAST=NOUSER1, na primer, ovi korisnici nisu u mogućnosti da vam šalju poruke.

Klasu USER2 takođe čine SEND poruke, ali poslate sa drugih nodova *Bitnet*-a.

Klasu USER3, najzad, čine sistemski odgovori na zahteve koji uključuju odzive servera i izveštaje o „propagaciji" vaše pošte kroz mrežu. Ukoliko, na primer, u svoj LOGIN.COM ugradite

$ SET BROADCAST=(USER1,USER2,NOUSER3)

dobijaćete sve poruke **osim** često zamornih izveštaja o tome kako vaša pošta putuje i gde sve stiže. Trenutno izabrane opcije ove zanimljive komande dobijate sa SHOW BROADCAST [Preuzeto sa Sezama, autor **Slobodan Marinković** (*slom*)]

475 [IGRE, AMIGA] Ako vam u sve popularnoj igri *Railroad* zafali gotovine, pritisnite F1 a onda Shift 4 i dobićete 500000 dolara. Ovo možete ponavljati do nekih 28 miliona dolara, a zatim će saldo „pobeći" u minus. Trik se može primeniti u bilo kojoj fazi igre, potreban broj puta. Otkriven je na Amigi, ali bi trebalo da funkcioniše i u verzijama ove igre za druge kompjutere [Preuzeto sa Sezama, autor **Dragan Micić** (*drami*)]

Priloge za ovu rubriku šaljite na adresu „Računari" (za „Bajtove lične prirode"), Bulevar vojvode Mišića 17, Beograd ili preko Sezama u vidu lične poruke uredniku (*mail write dejanr*)

NOVOGODIŠNJI PROBLEM

Novogodišnji zadatak je, kako su primetili i mnogi rešavači, bio prilično varljiv – pomenuti su tako veliki brojevi da je računarima pretilo usijanje, a u stvari se mogao „začas" matematički rešiti. Na ovaj ili onaj način, do magične brojke 9 došlo je preko 105 čitalaca „Računara" i korisnika Sezama, a našla su se i dva pogrešna odgovora...

Podsetimo se pre svega postavke – slovom A smo obeležili broj 1992^{1992} (broj 1992 podignut na stepen 1992), slovom B zbir svih cifara broja A, slovom C zbir cifara broja B a slovom D zbir cifara broja C. Trebalo je odrediti D.

Matematičko rešenje se zasniva na činjenici da je broj 1992 deljiv sa 3 (664·3=1992). To znači da ćemo stepenovanjem broja 1992 bilo kojim celobrojnim eksponentom takođe dobiti rezultat deljiv sa 3, a ako je eksponent barem 2, rezultat će biti deljiv i sa 9. Sa druge strane, znamo da je broj deljiv sa 9 ako i samo ako mu je zbir cifara deljiv sa 9, odakle zaključujemo da su i brojevi B (zbir cifara broja $A=1992^{1992}$), C i D deljivi sa 9.

Pokušajmo sada da grubo procenimo koliki su brojevi A i B. Da smo računali 9999^{2000}, kod svakog množenja dobili bismo broj sa „približno" četiri cifre više, što znači da bi posle 2000 množenja rezultat imao najviše 8000 cifara. „Naš" broj 1992^{1992} je očigledno dosta manji od ovih 8000 cifara, ali nam je za sada i ova procena sasvim dovoljna. Ako bi na svih 8000 mesta bile baš devetke, njihova suma (tj. broj B) bi bila 8000·9=72000 (videćemo da je i broj B dosta manji od 72000, ali nam i ova gruba procena „završava posao"), što znači da B ima najviše 5 cifara. Sve i da su te cifre devetke (što je nemoguće), njihova suma (tj. broj C) bi bila 45, odnosno dvocifran broj. Najveća moguća suma dva jednocifrena broja je 18 (9+9), ali je D manja od toga, a ne može biti 0, jer zbir cifara nekog prirodnog broja nikako ne može biti jednak nuli. Pošto znamo da je D deljivo sa 9, očigledno se radi baš o broju 9 – nikakav drugi broj između 1 i 17 nije deljiv sa 9! Ukratko, D=9.

Da smo ovaj zadatak postavili pre par godina (važilo bi, na primer, i za 1989, 1986...), može se zapravo pokazati da slično rezonovanje važi za bilo koji broj deljiv sa tri ako stepenovan samim sobom ima između 1112 i 11111 cifara, verujemo uostalom da je sličan zadatak zaista i postavljen na Republičkom takmičenju mladih matematičara neke od tih godina), izloženo rešenje bi bilo jedino moguće. No, od tada su personalni računari žestoko napredovali po pitanju karakteristika i nazadovali po pitanju cene, pa imamo priliku da rezonovanje proverimo i uz pomoć kompjutera. Uz pomoć nekog matematičkog programa (*Mathematica 386*, *MuMath*...) ili, na kraju krajeva, za ovu priliku specijalno napisanog programa sa slike 1 dobijamo punu vrednost broja A. Brojevi B, C i D su mnogo manji: 29745, 27 i, naravno, 9.

Najkompletnije rešenje ovoga zadatka dostavila je **Biljana Srdanov** iz Žagubice, kojoj pripada naša prva nagrada od 8000 dinara. Preostale nagrade smo izvukli, pa je 6000 dinara dobio **Ljubomir Josifovski** iz Skopja, a 4000 dinara **Branko Obradović** iz Novog Beograda. Nagradu za najbolje rešenje dostavljeno modemom dobija **Milovan Kovačević** iz Novog Sada, a čestitke za interesantna rešenja zaslužuju i *Milan Božić, Nenad Bunčić, Slavko Kukrika, Milenko Malović, Miorad Milivojša, Boban Simonović, Nina i Branislav Tomić* i *Jaroslav Uskoković*.

U ZNAKU BROJA 1992

Da još malo ostanemo u znaku godine koju polako „trošimo" – pronađite **najveći** prirodan broj koji je proizvod nekoliko prirodnih brojeva čija je suma 1992. Kao i prethodni, ovaj zadatak je predložio *Robert Paušić* iz Šida.

Rešenja Pitalice, zajedno sa **obrazloženim** programima koje ste napisali, pošaljite na uobičajenu adresu „Računari" (za Dejanove pitalice), Bulevar vojvode Mišića 17, Beograd) tako da pristignu **pre** 5. maja 1992. Sva pisma sa korektnim rešenjima konkurišu za novčane nagrade od 8000, 6000 i 4000 dinara, dok će kuponi (ili njihove fotokopije) na koja je upisan identifikacioni broj učestvovati u godišnjem takmičenju rešavača Pitalica. Identifikacioni broj dobijate tako što u prve tri kućice upišete **poslednje** tri cifre nečijeg broja telefona, u sledeće dve godinu vašeg rođenja i na kraj dodate dve cifre po izboru. Obratite pažnju da sva rešenja koja šaljete u toku godine nose isti identifikacioni broj. Osim običnom, rešenje ove Pitalice možete da pošaljete i elektronskom poštom posredstvom Sezama – koristite meni REDAKCIJA PITALICE REŠENJE ili iz komandnog moda pošaljite ličnu poruku korisniku pitalice (*mail write pitalice*). Sva rešenja pristigla elektronskom poštom ravnopravno sudeluju u izvlačenju nagrada i konkurišu za **specijalnu nagradu** od 8000 dinara, koja će biti dodeljena najboljem rešenju koje je dostavljeno modemom!

```
74: Najveći prirodan broj koji je proizvod nekoliko prirodnih
    brojeva čija je suma 1992 je ______________________

Ime i prezime  __________________________________________
Adresa         __________________________________________
Mesto          __________________________________________
Kompjuter      ____________________  Vreme  _____________

Identifikacioni broj:  |_| |_| |_|   |_| |_|   |_| |_|
                       iz broja tel.  god. rođ.  po izboru
```

Listing 1

```
program 1992;
{
          Rešenje 72. Pitalice
   prema programu Milovana Kovačevića
            "Računari 81"
}
var
  cifre:array [1..7000] of integer;
  i,j,c,d,brojcif,prenos:integer;
  b:longint;

function zbir(x:longint):integer;
var
  i,z:integer;
begin
  z:=0;
  while x>0 do begin
    z:=z+x mod 10;
    x:=x div 10;
  end;
  zbir:=z;
end;

begin
  for i:=1 to 7000 do cifre[i]:=0;
  cifre[1]:=2;
  cifre[2]:=9;
  cifre[3]:=9;
  cifre[4]:=1;
  brojcif:=4;
  for i:=2 to 1992 do begin
    prenos:=0;
    for j:=1 to brojcif do begin
      cifre[j]:=cifre[j]*1992+prenos;
      prenos:=cifre[j] div 10;
      cifre[j]:=cifre[j] mod 10;
    end;
    while prenos>0 do begin
      inc(brojcif);
      cifre[brojcif]:=prenos mod 10;
      prenos:=prenos div 10
    end;
  end;
  b:=0;
  for j:=1 to brojcif do b:=b+cifre[j];
  c:=zbir(b);
  d:=zbir(c);
  writeln;
  writeln('d = ',d);
  writeln
end.
```

Program za izračunavanje broja A

Tokom februara smo „opslužili" 19 397 poziva, komunicirali sa korisnicima 4 282 časa, tj. 178 dana (obzirom da je ovaj prestupni februar imao 29 dana, u proseku je bilo neprekidno zauzeto 6 nodova), od čega je u chat-u provedeno 1064 časa, tj. 133 radna dana (noći). Korisnici su sa Sezama download-ovali 777 megabajta datoteka, dok je za islo vreme sistemu poslato oko 52 megabajta.

U javne konferencije je prostiglo 7 338 poruka (5,9 megabajta teksta) i 4 megabajta datoteka priključenih uz njih, 5672 privatne poruke (preko dva megabajta privatne pošte) i 2 590 poruka u grupama (860 kilobajta teksta), ne računajući datoteke priključene uz te poruke.

U konferenciji **SEZAM** bilo je nešto manje poruka nego prethodnih meseci, pre svega zato što nije bilo značajnijih izmena u Sezamovom softveru ili SOR-u. Poruke su se uglavnom svodile na predloge za sledeću „generalnu reviziju" SOR-a koja se planira za sledeće mesece, a i na diskusiju o raznim „aktuelnim događajima" na Sezamu, novim temama koje bi trebalo ili ne bi trebalo pokrenuti, konferencijama koje se bliže svom limitu od 5000 poruka i tome slično.

Konferencija **RAČUNARI** ima već ustaljen dnevni (tj. mesečni) red. Sredinom februara najavljen su novi „Računari" a onda se diskutovalo o njihovom sadržaju i (naravno) ceni.

U konferenciji **KOMUNIKACIJE** najviše je bilo reči o popravci modema, napravi čiji su releji često projektovani kao povremena zamena za (kod nas i dalje retko) tonsko biranje, pa imaju nezgodan običaj da pregore. Na sreću, zamena se obično može naći i u našim radnjama, saznali smo i u kojim. Bilo je i (već tradicionalne) diskusije o dobrim i lošim stranama komunikacionih programa, a pojavila se i nova zvezda – program Odyssey koji, kada jednom uspete da ga instalirate, nudi bolju MNP emulaciju od poznatog programa MTE. Počeli smo malu diskusiju o hakerima u istoimenoj temi, ali začudo diskusija nije dugo „živela". Što se velikih sistema i elektronske pošte tiče, govorilo se o Internet-u i bazama koje nudi, (eventualnom) pristupu ovoj mreži iz Jugoslavije, problemima sa Bitnet servisom, arhiverima za VAX i naročito o DECOR-u, prvom domaćem off line reader-u za VAX koji je specijalno za korisnike Sezama napisao Dragiša Dunc (dragisha).

U konferenciji **PC HARD** ponovo je bilo puno poruka i to još jednom zahvaljujući Stacker-u, programu koji znatno uvećava kapacitet diska tako što „u letu" arhivira i dearhivira datoteke. Opisivani su neki problemi na koje korisnici Stacker-a nailaze na nekim specifičnijim konfiguracijama. Diskutovalo se i o promeni BIOS-a, pošto svaka 386/486 ploča očito ne prima svaki 386/486 BIOS – šta se može zameniti, a šta ne može? Bilo je, naravno, i pravih hardversko-tržišnih diskusija: o cenama u USA i mogućnosti uvoza opreme poštom, razlikama između 80486 ploča na 25, 33 i 50 megaherca, ugradnji eksterne baterije za časovnik realnog vremena (zvuči jednostavno, ali ponekad baš i nije), upotrebi Suntec AT čipova, novom setu drajvera za ET4000 VGA (upravo se pojavio u USA), ugradnji YU slova u NEC P60 i Star štampače.

Što se konferencija **PC OS** tiče, u centru pažnje je bila tema **ms dos** i diskusija o povećanju primarne DOS particije sa „klasičnih" 32 megabajta na ukupnu veličinu diska, naravno bez gubitka podataka. Na žalost, konačni zaključak je bio da je to praktično nemoguće. U okviru teme **unix** se, između ostalog, pisalo o PD verzijama UNIX-a, PD softveru za UNIX, arhiverima itd. Tema **windows** je primila najveći broj poruka – većina je bila vezana za fontove (konverzija između različitih formata, naša slova itd), a počeli su da pristižu i PD i shareware programi za Windows okruženje. Najzanimljivije diskusije u temi **4.dos** se odnosile na probleme koji nastaju zato što je datoteka 4DOS.COM u novoj verziji ovog paketa duža od 64K, pa je morala da postane EXE. Pomenimo prve iskustva (i, naravno, probleme) sa operativnim sistemom DR DOS 6.0, koja su opisana u okviru teme **razno**.

Februar je u konferenciji **PC PROG** počeo problemom skraćivanja postojećeg fajla bez njegovog prepisivanja, ali je ipak najzanimljivije bilo kada su programeri kroz ovu diskusiju priznali retko izgovorenu istinu da programi nisu 100% sigurni, već da rade sa „velikom verovatnoćom". Moramo primetiti i nagli porast interesovanja za neke nestandardne stvari, kao što je programiranje procesora 6502, 65C02, 8080 i Z80, a paskal programeri su odlučili da načas zaborave Turbo Pascal i pozabave se generisanjem slučajnih brojeva na standardnom paskalu, pa čak i na Mac Pascal-u. U okviru drugih tema uglavnom su se vodile diskusije o problemima i nepoznanicama u onim stvarima za koje smo svi mislili da su nam odavno do kraja poznate. Tako su clipper programeri diskutovali o dbedit-u, tbrowse-u i get sistemu, oni koji koriste C o for() funkciji, bejzik programeri o 64K limitu itd. Ipak, februar će biti zapamćen i po živoj diskusiji o zaštiti programa – bilo je mnogo predloga, počev od onih da program proverava postojanje nekog fajla, setovanja, provere atributa, provere dužina datoteka itd.

Najzapaženija diskusija u konferenciji **PC USER** vodila se oko drajvera za tastaturu. Goran Bročić i Dragan Zakić (poznati kao autori programa koji rešavaju problem YU slova na VGA i Herculesu) zakazali su čak i „dvoboj" svojih drajvera koji se, izgleda, završio bez pobednika. Diskusija iz prethodnog meseca o prednostima i manama postojećih tekst procesora se u februaru (doduše u manjem obimu) prenela na editore – izgleda da je najomiljeniji editor na Sezamu QEdit, ali ni Brief se ne predaje lako. Diskutovalo se o WordPerfect-u, ChiWriter-u (uglavnom o fontovima sa YU slovima), TeX-u. Po zatvaranju stare konferencije KOMUNIKACIJE, tema **arhiveri** je premeštena u konferenciju PC USER, gde su odmah nastavljene stare diskusije, uglavnom inicirane novim verzijama PKZIP i ARJ-a. U okviru teme virusi je (logično) bilo najviše reči o „dolazećem" Mikelanđelu – sezamovci ga dočekuju dobro opremljeni antivirus programima iz Sezamovih direktorijuma, a izveštaje iz „bitke" očekujte u sledećem Biltenu. Ostali razgovori su bili manje-više standardni – o fontovima za Venturu i Corel Draw, CAD programima, nabavci softvera.

I ovog meseca **AteriSTi** su najviše pričali o igri OXYD 2 – možda zvuči čudno da jedna igra bude glavni predmet razgovora puna 2 meseca, ali nije lako preći 100+100 nivoa. Od „ozbiljnih" stvari, najaktivnije je bilo u temi **emulatori**, gde se pored, sada već klasičnog Spectrum emulatora, pričalo i o Macintosh-u tj. Spectre GCR za koju je poslato dosta korisnog softvera. Tema „komunikacije" je imala dosta poruka o tome kako podesiti eksterni ZMODEM FTP sa pojedinim komunikacionim programima i o eventualnoj mogućnosti rada ZMODEM-a u pozadini. Probleme je zadavala i nova verzija AHDI-ja 4.0 (Atari HD drajver) koja je zauzimala više memorije nego prethodna, pa neki programi ne mogu da rade ne mašinama sa 1Mb RAM-a.

Konferencije **ORKA** ni ovoga puta nije mogla da prođe bez diskusije o virusima – jedan radoznali Sezamovac je namerno zarazio svoj disk virusom Michelangelo i podesio datum na šesti mart, dan aktiviranja virusa. Njegov zaključak: „Nemojte da se šalite sa Mikelanđelom!" Ponovo se aktivirala tema najbolji, ali ovoga puta ne na relaciji PC Atari nego po pitanju korisničkih interfejsa: komandna linija ili prozori i menji? Tema **elektron** je i dalje vrlo aktivna: razvila se rasprava o najboljim pojačalima, upotrebi transformatora za zvonce za napajanje eksternih modema itd. Saznali smo, u okviru teme **zakoni**, nešto više o novim porezima na autorske honorare. I tokom februara najaktivnija je bila tema **igre**: traženi su i davani saveti za mnoge, što stare, što nove PC igre. Korisnici HP48SX su najzad dobili precizne podatke o potrošnji struje njihovog mezimca u raznim režimima rada.

U konferenciji **FORUM** se tokom februara, pored diskusija o aktuelnim događajima, vodila veoma žustra rasprava o odazivanju ili neodazivanju na mobilizacijske pozive. Govorio se i o (tada približavajućem) 8. martu, kao i o tretmanu koji različiti događaji dobijaju u medijima, posebno na televiziji. Prenošene su neke političke diskusije sa hrvatskih i slovenačkih BBS-ova, što je bila dobra prilika da se, doduše pasivno, uporede stavovi koji preovlađuju „ovde" i „tamo". Bilo je opštih tračeva u temi **trač** i posebnih tračeva u temi **devojke**, a ne treba zaboraviti ni neočekivano žestoku diskusiju o pravima kućnih ljubimaca koji sve češće stradaju na ulicama, kao i problemima koje ljudi imaju sa neočekivano „podivljalim" (nekada kućnim) životinjama.

Februar je u konferenciji **EXTRA** protekao je u znaku FEST-a o kome se detaljno diskutovalo u okviru teme **pokretne slike**. Mnogo poruka bilo je i u temama **misterije i zvezde**, najviše o odnosima između astrologije i astronomije. Izgleda da u ovo krizno vreme astrologija pobeđuje astronomiju – niko više i ne primećuje kada se u ozbiljnim informativnim emisijama Astronomsko društvo „Ruđer Bošković" naziva „astrološkim društvom". Bilo je dosta poruka u temi **automobili** (tradicionalna diskusija o prednostima i manama nemačkih automobila u odnosu na, recimo, italijanske), dok su **hi fi** i **sport** bile standardno posećene.

U konferenciji **VICEVI** bilo je zabavno kao i uvek. Novi vicevi pristižu svakodnevno, a posebno je zanimljivo bilo kada su korisnici počeli da šalju razne „bisere" nastavnika (a i učenika) iz raznih škola. Bilo je zaista izjava za istoriju! Glasanje za najbolji vic januara je malo kasnilo, ali ne toliko da ovde ne bismo mogli da objavimo najkraći od pobedničkih viceva: Kakva je razlika između morskog praseta i žene inženjera? Nema razlike: ni morsko prase – niti je morsko, niti je prase. Uskoro počinje i glasanje za najbolji vic 1991. godine, a u konkurenciji su svi vicevi pobednici mesečnih takmičenja u prošloj godini.

KAKO POSTATI ČLAN

Postupak učlanjenja počinje zvanjem Sezama. Nakon poziva broja 648-808 i uspostavljanja veze (parametri 2400N8) i pozdravne maske, dobićete pitanje:

Username

Na njega treba odgovoriti sa **NEW** (novi korisnik), nakon čega vas Sezam vodi kroz proces prijave. Prvi podatak koji treba da date je vaše puno ime i prezime, a potom birate pseudonim pod kojim ćete koristiti sistem. Pseudonim je bilo kakva reč do 16 znakova koja sadrži samo mala slova i znak _ (tačka) i obično se bira tako da asocira na vaše ime. Sistem će na osnovu vašeg pravog imena predložiti pseudonim koji možete ali i ne morate prihvatiti. Pre izbora pseudonima dobro razmislite, jer on postaje vaše trajno ime na Sezamu i ne možete ga više promeniti.

Sledeći korak je izbor lozinke (password) kojom se vaš pseudonim štiti – prijavljivanje sistemu je moguće samo ako poznajete lozinku koja je vezana uz vaš pseudonim.

Nakon ovih osnovnih podataka sledi izbor nekoliko radnih parametara (objašnjenje ćete videti na ekranu) a potom unos ostalih ličnih podataka – adresa, telefona i podataka o opremi koju koristite za zvanje Sezama. Ovim je prva faza gotova.

Posebno obratite pažnju da **adresu unesete korektno** – ispravna adresa je jedini način da Uprava ostvari kontakt sa vama, kako bi se do kraja obavio proces vašeg učlanjenja.

Sledećeg radnog dana će vam biti poslato pismo u koma ćete pronaći i popunjenu uplatnicu. Na vama je da izaberete period na koji se učlanjujete i na osnovu toga na uplatnicu uplatite iznos pretplate:

za 1 mesec	**1800 din**
za 3 meseca	**3500 din**
za 6 meseci	**6300 din**
za godinu dana	**10000 din**

Nakon izvršene uplate, fotokopiju potvrde o uplati pošaljite na naznačenu adresu Uprave Sezama ili je pošaljite (daleko brže) faksom. Po prijemu ove potvrde, Uprava će vam istog radnog dana odobriti pristup sistemu, čime postaje punopravan član Sezama. Istog dana će na vašu adresu biti poslato Uputstvo za korišćenje sistema.

Ukoliko pri prvom prijavljivanju imate problema i ne završite popunjavanje podataka korektno, nije potrebno da zovete ponovo i prijavljujete se pod novim imenom. Sačekajte jedan dan pa pozovite ponovo – administratori sistema kod detektovanja nepotpunih prijava interveniše i daju vam mogućnost da pri sledećem javljanju ponovo popunite prijavu ili vam ostavljaju ličnu poruku kojom vas obaveštavaju šta treba dalje da učinite.

Ukoliko ste se ipak prijavili pod više različitih pseudonima, imajte na umu da Uprava sama mora da izvrši izbor koju od viših prijava će uzeti u dalju obradu. Kriterijum je najčešće kompletnost podataka. U slučaju više korektnih prijava uzima se u obzir samo poslednja. U svakom slučaju, na uplatnici će biti upisan pseudonim na koji se uplata odnosi – samo njega možete koristiti.

Ukoliko želite pre prijavljivanja da razgledate sistem, na pitanje Username odgovorite sa DEMO. Naći ćete se u demonstracionom režimu rada u koma možete pregledati kompletnu strukturu sistema, koja je identična onoj „pravoj". Broj DEMO poziva nije ograničen a vreme za jednu sesiju je 5 minuta.

VIRUSI

U vreme paničnog straha od računarskih (ili kompjuterskih, kako ko voli) virusa, nije nikakvo iznenađenje što se i na Sezamu najviše traže programi za njihovo otkrivanje i uništavanje. I dok su nas sa stranica dnevnih novina plašili laičkim tekstovima o nadolazećoj "kataklizmi" u vidu virusa "Mikelanđelo" (*Michaelangelo*), korisnici Sezama su mirno čekali "sudnji dan", jer im je već mesecima bio dostupan antivirusni program sa kojim su bili bezbedni. Reč je o programu *SCAN* (i njemu srodnim antivirusnim programima kompanije "McAfee"), koji je sposoban da otkrije (a *CLEAN* da uništi) taj vrlo opasni virus još od verzije 7.9v84 (izašle oktobra 1991.)

Usled velike dinamike pojavljivanja novih virusa i njihovih klonova, i "McAfee" vrlo redovno izbacuje nove verzije njihovih programa. *Scan* verzija 8.3B86, koja se pojavila u februaru, sposobna je da otkrije sva (bar je tako naznačeno u dokumentaciji) poznate viruse, kojih na spisku ima čak 1199, uključujući i razne podvarijante i klonove. Značajna je novouvedena opcija **/AG**, koja će na kraj svake EXE i COM datoteke (slično **/AV** opciji) dodati informacije dužine 52 bajta, koje kasnije može koristiti *Clean* da bi uspešno popravio oštećene datoteke. **Scan /AG** će takođe sačuvati i kopije *boot* sektora i particione tabele, donoseći time dodatnu sigurnost u zaštiti od virusa. Ekvivalentno **/AV** opciji, uz **/AG** postoje i **/CG** (za proveru sačuvane informacije i otkrivanje infekcije) i **/RG** (za uklanjanje tih podataka).

NOVE DATOTEKE

Uz *Scan & Clean* najzapaženije su pojave nove verzije *ARJ* arhivera (koju ćemo malo opširnija opisati) i *SHEZ*-a, programa za rad sa arhiviranim datotekama. Dobili smo i nešto novu verziju programa *800* II 1.71 od one koja je ranije bila na Sezamu (II 1.40) i u kojoj su ispravljeni neki manji problemi.

Direktorijum **\IBMPC\INFO**, namenjen uglavnom raznoj dokumentaciji, u februaru je doneo nekoliko vrlo korisnih noviteta – podatke o svim "Seagate" hard diskovima, tehničke informacije o "Borland"-ovim kompajlerima, kao i novu verziju liste sa podacima o svim PC interaptima. Brojni vlasnici "Seagate" hard diskova su sigurno našli korisne podatke u datoteci **\ibmpc\info\1seagate.zip**. Ona sadrži datoteke sa brojnim tehničkim podacima za oko 130 raznih modela, uključujući tu i poslednju seriju sa kraja 91. Uz opis konektora i prekidača i sve moguće karakteristike, data su i korisna uputstva za instalaciju diskova (kako ih podesiti u CMOS RAM Setup-u i formatirati).

Na Sezamu su se našle i zbirke tehničkih informacija za "Borland"-ove Turbo Pascal i C kompajlere. Podeljene su po pojedinačnim temama (ima ih ukupno po oko 100 za C i TP) i uglavnom se sastoje od (nekad i vrlo opširnog) teksta koji opisuje problem i rešenje, kao i pratećeg listinga. Zbirka za C pokriva *Borland C++ 2.0*, *Turbo C++* i starije (Turbo C) kompajlere, dok su informacije za *Turbo Pascal* nešto starije i pokrivaju verzije zaključno sa TP 5.0. Ove datoteke se na Sezamu nalaze u direktorijumu **\ibmpc\info** kao **bchelp10.arj** (za *Borland Turbo C*) i **tpti10.arj** (*TP*).

	Tabela 1: TOP 10 Datoteka – Februar 1992.			
	direktorijum	datoteka		opis
1	\IBMPC\VIRUS	scanv86b	zip	SCAN – McAfee virus detektor 8.3B86
2	\IBMPC\VIRUS	clean86b	zip	CLEAN – Uklanja viruse koje Scan nađe
3	\IBMPC\ARCHIVER	arj230	exe	ARJ arhiver (v 2.30)
4	\IBMPC\ARCHIVER	shez65	zip	Shell za rad sa zip, arj, lha i ostalim arhiverima
5	\IBMPC\UTILITY	800ii171	zip	Korišćenje disketa sa većim kapacitetom od DOS-ovog
6	\RSOFT	r078cide	zip	Clipper demo programi
7	\IBMPC\VIRUS	vshld86b	zip	Rezidentni virus detektor
8	\COM	sor230	zip	SOR: Sezam Off Line Reader
9	\IBMPC\YU	hyutil20	zip	Hercules YU utilities (dzakic)
10	\IBMPC\YU	wp_yu	zip	Word Perfect 5.1 YU utilities (dmiric)

WINDOWS direktorijum se i dalje puni programima slabije nego što smo to očekivali. I pored brojnih priloga korisnika (koje moramo pohvaliti na uloženom trudu), veliki broj novih programa za *Windows* ipak ne zadovoljava neke naše (nadamo se ne suviše visoko postavljene) kriterijume za smeštanje nekog programa u Sezamove direktorijume. U proteklom periodu smo videli veliku količinu uglavnom prilično nepotrebnih stvari, ali se ipak nadamo da će se iz tog mnoštva ubrzo izdvojiti kvalitetni programi, a jednog dana i posebna grana WINDOWS direktorijuma. Tako su se u februaru u **\ibmpc\windows** direktorijumu našli "samo" (vrlo relativan pojam) Windows verzija (v 17.1) fraktal generatora *Fractint*, nova verzija *WinQVT* komunikacionog programa i program za editovanje .INI datoteka.

Programi za rad sa našim slovima, uglavnom mali ali vrlo korisni, mogu se naći u **YU**, tačnije **\ibmpc\yu** direktorijumu, koga još uvek nemamo nameru da preimenujemo, niti da otvaramo posebne SER, CRO, SL... poddkataloge. Novosti u njemu su bile nova verzija super brzog programa za konverziju tekstova iz jednog u drugi YU kodni raspored **Dragana Zakića (dzakic)** (ranije je ovo bio deo paketa *HYUtil*) i *Word Perfect 5.1 YU Utilities* – zbirka makroa za WP koja obezbeđuje sve što vam je potrebno da bez problema koristite naša slova. Autor *WP_YU utilities* je **Dragan Mirić (dmiric)**.

Jedna od najtraženijih datoteka u februaru je bila **r078cide.zip** (Clipper listinzi uz tekst "Programi koje rade sami" **Nenada Batocanina**) iz **\RSOFT** direktorijuma. Ovde možete naći (skoro) sve listinge uz članke koji se pojave na stranicama "Računara", a često i mnogo više nego što može da stane na papir.

Tabela 2 Novo u februaru i ukupno				
	FEBRUAR		UKUPNO	
direktorijum	broj	dužina	broj	dužina
ARCHIVER	2	386 306	16	1 139 156
COM	1	161 052	28	2 734 410
DEMO	3	269 972	11	1 122 568
DRIVERS	1	87 047	15	1 553 570
EDITOR	1	135.552	13	926 198
FUN	0	0	15	1 362 719
GRAPHICS	0	0	11	994 992
INFO	6	942 411	14	1 781 227
MISC	1	25 261	22	2 188 958
PROGRAM	1	74 388	22	2 380 221
SYSTEM	1	166 194	15	1 208 139
TEST	1	54 076	10	784 435
UTILITY	5	326 525	34	1 481 916
VIRUS	5	579 218	9	729 024
WINDOWS	4	436 637	10	1 136 802
YU	2	15 922	11	150 914
Ukupno	34	3 660 563	256	21 797 249

Da nabrojimo na brzinu i ostale novitete – nove verzije kompajlera BAT datoteka (*BAT2EXE* iz "PC Magazine"-a), *Hyper Disk* keš programa (verzija 4.31 uz minorne izmene u odnosu na prethodnu), Super VGA BGI drajvera, VDE tekst editora (v 1.62b), *TeleReplica* programa za *remote* kontrolu računara (v 3.18).

ARJ vs ZIP

U vreme kada smo prošle godine (u "Računarima 73") testirali najpopularnije PC arhivere (ili programe za kompresiju datoteke, na Sezamu je bilo rasprava koji termin je ispravno koristiti), bili smo oduševljeni ARJ-om. Najvažniji nam je bio stepen kompresije, koji je bio skoro uvek dovoljno bolji od ostalih arhivera, uključujući tu i PKZIP, ali ARJ je svoje mesto na Sezamu (kao poluzvanični standard pored ZIP-a) našao i zbog nekih svojih jedinstvenih mogućnosti – opcija podela arhiva na delove željene dužine (-v) nam je uštedela dosta muke prilikom podele dugačkih datoteka i pravljenja bekapa. Tada je najveća (i jedina ozbiljna) zamerka ARJ-u bio relativno spor rad. Od tada je došlo do ne baš malih promena. Najznačajnije su obznanjene pojavama novih verzija arhivera – *PKZIP 1.93 ALPHA* (test verzija) i *ARJ 2.30*. *PKZIP* je u toj beta (tačnije alfa) varziji doneo veliko poboljšanje performansi, tj. i stepena kompresije i brzine – po prvom se potpuno približio *ARJ*-u, a po drugom ga debelo prestigao. Ovo je verovatno podstaklo autora *ARJ*-a da svom arhiveru poboljša pre svega stepen kompresije, uz neke minimalne (varljive) dobitke na brzini (a najavljena su i takva poboljšanja u planiranoj verziji 2.40). Na žalost, od pojave test verzija (koja se zbog potencijalne nepouzdanosti ne preporučuje za redovnu upotrebu) *PKZIP*-a (15. oktobra 91. – tačno godinu dana posle pojave verzije 1.1) do sade (početkom marta) još uvek nismo dobili onu koju bismo mogli bezbrižno da koristimo. S obzirom da se na polju programa za kompresiju podataka očekuje još dinamičnog razvoja, vrlo je nezahvalno preporučivati konkretni arhiver. Mi smo i dalje privrženi *ARJ*-u, koji je do pojave "prave" verzije *PKZIP*-a ipak najbolji pouzdan izbor.

Sve arhivere i korisna alatke vezana za njih na Sezamu možete naći u **\ibmpc\archiver** direktorijumu.

Zoran Životić

#include

H datoteke su, iako ne spadaju u sam C jezik, sigurno jedan od najlakših znakova raspoznavanja C programa – posebno direktiva #include <stdio.h> koja se tako često i bez razmišljanja kuca. Ovoj uvodnoj datoteci možda jedino malloc može da konkuriše po popularnosti, pre svega zato što njeno izostavljanje, zbog deklarisanja n za funkcija koje ne vraćaju int često ima loše posledice po program. Manje popularna, ali iz istog razloga važna je i string.h. Ponekad po navici uključuje i stdlib.h jer se u njoj nalaze deklaracije niza „svakodnevnih" funkcija. Ako ne znate gde je neka funkcija ili konstanta deklarisana, velika je verovatnoća da je baš u stdlib.h.

C je standardno opremljen sa oko tri desetak H datoteka, pa se postavlja pitanje kako znati kada koju od njih, pored pomenutih očiglednih primera, treba uključiti u program. Pitanja koja uz to ide je i da li je baš stvarno potrebno uvek uključiti sva zaglavlja u kojima se pojavljuju deklaracije svih konstanti i bibliotečkih funkcija koje se javljaju u programu.

Odgovor na ovo drugo pitanje je – jeste. Tri su razloga za ovako kategoričan odgovor, iako ćete u praksi naići na situacije da vaš program savršeno funkcioniše a da pri tome ni neznate da li ste uključili sva potrebna zaglavlja. Prvi razlog je najočigledniji – konstanta koje koristite u programu a nisu deklarisane zbog toga što niste uključili odgovarajuće zaglavlje kompajler tretira kako promenljive, naravno kao nepostojeće promenljive i od generisanja izvršnog koda neće biti ništa.

Drugi razlog su već pomenute deklaracije funkcija koje vraćaju neki drugi tip podataka a ne int. Ovo je naročito važno kada radite u memorijskom modelu sa far pointerima. Rezultat malloc funkcije će u ovim memorijskim modelima biti prosto iskasapljen – pointeru dodeljen samo ofsetni deo vrednosti, što sigurno ne može dobro završiti. Doduše, ovakve vrste grešaka možete dosta lako uočiti ako zadate da kompajler izdaje više upozorenja od standardnih. Kod MSC ovo se postiže prekidačem –W2 ili –W3 (warning level), nakon čega će biti prijavljene sve funkcije koje se koriste a nisu deklarisane, sve dodele nekompatibilnih vrednosti (int u pointer i slično) kao i sve konverzije nakon kojih se gubi preciznost rezultata (long u int, far pointer u near pointer i slično).

Treći razlog je najmanje uočljiv ali podjednako važan. Radi se o takozvanom implicitnom promeni tipa (implicit cast) koju kompajler sam obavlje ukoliko ima dovoljno informacija da to učini. Evo jednog primera. Pretpostavimo da koristite najniži nivo rada sa datotekama preko _dos_read/_dos_write funkcija i da je vaš program u malom memorijskom modelu. Ukoliko ne uključite zaglavlje dos.h u kojoj je deklaracija ovih funkcija, a u programu se pojavi

_dos_read(handle, buffer, readbytes, &result)

program će sigurno raditi sve samo ne ono što ste zamislili. Kada se zaglavlje dos.h uključi problem nestaje. Razlog se krije u činjenici da funkcija _dos_read (kao i _dos_write) očekuje da promenljiva buffer bude far pointer, kao što je to zapisano u zaglavlju

_dos_read(int, void far *, unsigned, unsigned *);

Ukoliko kompajler nema ovu informaciju, na stek će preneti near pointer i eto problema. Ako tačno zna šta je potrebno, kompajler će generisati odgovarajući cast, kao da ste sami napisali

_dos_read(handle, (void far *)buffer, readbytes, &result)

Nema nikakvog razloga da očekujete da jedan od parametara neke funkcije iskače iz opšteg šablona koji važi za određeni memorijski model – zato je važno da uključite odgovarajuće zaglavlje, kako o takvim stvarima ne bi morali da vodite računa.

Konačno, da odgovorim i na ono prvo pitanje – kako znati kada je koja H datoteka potrebna? Na žalost, na ovo pitanje se ne može odgovoriti nekim opštim i jednostavnim šablonom – potrebno je prosto naučiti kada šta ide. Najbolji način da do ovog znanja dođete je da uključite viši nivo upozoravanja kompajlera (–W3 kod MSC) i da ne ignorišete ni jedno upozorenje koje dobijete. Čim vidite upozorenje da se funkcija xxx koristi a nema deklaraciju, odmah potražite u dokumentaciji u kom zaglavlju se nalazi (QH je najbrži način da dođete do ove informacije).

Ovakav način rada će vam možda u početku



smetati jer ćete dobijati i dosta upozorenja za koja ćete biti sigurni da su potpuno bezopasna, naročito onih koja govore o tome da vaše funkcije nemaju odgovarajuću deklaraciju. Propisno deklarisanje je svakako zamorno, ali i praksa koja se ipak isplati.

Za #include direktivu je inače vezana i jedna manje poznata zanimljivost. Svakako vam je poznata razlika između navođenja naziva datoteke u okviru znakova manje-veće (<stdio.h>) i znakova navoda ("stdio.h"). Prvi slučaj govori da datoteku treba pronaći u sistemskom direktoriju samog kompajlera, bilo da je definisan –I prekidačem bilo sa SET INCLUDE. MSC omogućuje da se INCLUDE promenljiva postavlja poput PATH-a – dozvoljeno je navođenje više direktorija odvojenih znakom ;.

U drugom slučaju se najčešće misli da preprocesor traži datoteku u tekućem direktoriju, što je pogrešno – datoteka se traži u direktoriju u kome se nalazi sam program. Dakle, ako zadajemo naziv programa sa stazom, ista staza će biti upotrebljena i za pronalaženje datoteke zaglavlja. S obzirom da se zaglavlja mogu pozivati jedno unutar drugog, tada pretraživanje ide u više koraka – prvo se datoteka traži tamo gde je datoteka u kojoj se nalazi #include direktiva, potom se koristi jedan korak nazad i traži po direktoriju u kome se nalazi datoteka u koju je uključena ova koja se trenutno obrađuje i tako redom, korak po korak unazad, do direktorija u kome se nalazi osnovna datoteka od koje je sve u počelo. Tek ako je ni tu ne pronađe, preprocesor odustaje.

Prioritet operatora

Možete li da zaključite koliko puta će biti izvedena sledeća petlja

```
x = 7;
for( ; ; )
{
    if (x&1 == 0) break;
    x >>= 1;
}
```

Tačan odgovor na ovo pitanje nije „3 puta" već – „dok ne ugasimo ili resetujemo računar". Greška se krije u uslovu izlaska iz petlje koji, kada se zagradama naznači redosled kojim ga kompajler ovako napisanog sređuje, izgleda ovako

if (x & (1 == 0)) break

S obzirom da je rezultat izraza 1 == 0 uvek nula, to ni x&0 nema šanse da kada bude tačno. Prioritet operatora u C nije stvar koja se može odmah na prvi pogled zapamtiti. Zato ćete u C programima najčešće sresti višak zagrada u izrazima, što je sasvim preporučljiva praksa. Ipak, određeni broj zagrada možete lako eliminisati ako zapamtite par pravila.

Prvo pravilo sigurno znate jer i ne deluje da se radi o operatorima – najviši prioritet (isti kao i zagrade) imaju operatori . i ->. Dakle, adresiranje polja strukture bilo direktno (struktura.polje), bilo preko pointera (struktura->polje) je dovoljno čvrsto vezano da se može bez opasnosti koristiti bez ikakvih zagrada u svim izrazima.

Sledeći niz prioritet imaju operatori nad jednim operandom, kao što su ++, --, ! itd. I ovde možete bez razmišljanja izbaciti zagrade jer se i oni čvrsto drže uz svoj operand, pa bi ih samo veštačkom zagradom mogli odvojiti.

Nakon ovoga dolaze aritmetički operatori (+, –, *, /, %) koji imaju sasvim logičan prioritet, ali nakon njih zalazimo u oblast binarnih i logičkih operatora gde svaka jasna logika prestaje. Recimo, operator < ima veći prioritet od operatora !=, binarni operator ima veći prioritet od operatora ! ali manji od & itd. Zato je najjednostavnije zapamtiti dno ove liste prioriteta, a sve međuslučajeve razjasniti upotrebom zagrada.

Na dnu liste su dva logička operatora – && i || što znači da složene logičke izraze možemo bezbedno pisati bez zagrada, na primer

if (x!=10 && y>5)

I ispod ovoga postoje operatori sa nižim prioritetom, ali su oni, kao pomenuta dva sa najvišim, poznati jer se radi o operatorima dodele (=, +=, -=, >>= itd) pa je sasvim logično da će ceo izraz sa desne strane biti sređen pre nego što se konačna vrednost dodeli promenljivoj sa leve.

qsort

Efikasni algoritmi za sortiranje su večna tema, ali ima i sortiranja kod kojih efikasnost nije mnogo bitna. U programu kupite listu datoteka tekućeg direktorija i želite da je sortirate pre ispisa – pomisao da sada treba napisati 10-20 linija koda da bi se ovaj trivijalan posao obavio ne deluje baš privlačno. Na sreću, deo standardne C biblioteke je i funkcija qsort, koja je idealna za ovakve situacije. Funkcija je realizacija poznatog quick sort algoritma, što je čini primenljivom i u nešto složenijim slučajevima, kada očekujemo da lista za sortiranje može biti i duža. Deklaracija funkcije (iz stdlib.h) je

void qsort(void *, size_t, size_t, int (*)(void *, void *))

Za funkciju je karakteristično da niz koji se sortira treba da bude unutar kontinualnog memorijskog bloka (početna adresa niza je prvi parametar), da se zadaje ukupan broj članova niza i dužina jednog elementa i konačno, pointer na funkciju koja treba da obavi poređenje dva elementa niza. Deklaracija deluje malo komplikovano zbog onog poslednjeg parametra, ali će nakon sledećeg primera verovatno biti jasnija.

```
char data[MAX_ELEM][ELEM_SIZE];
qsort(data, MAX_ELEM, ELEM_SIZE, strcmp);
```

S obzirom da sortiramo stringove, kao funkciju za poređenja smo iskoristili bibliotečku funkciju strcmp. Uslov da funkcija može biti upotrebljena je da kao parametra koristi pointere na element niza (ovi pointeri se prosleđuju iz same qsort funkcije) i da vraća

- negativnu int vrednost ako je prvi element manji,
- 0 ako su jednaki i
- pozitivnu int vrednost ako je prvi element veći od drugog.

U slučaju da želimo da sortiramo niz struktura koje smo sami deklarisali, onda se qsort koristi na sledeći način

```
typedef struct direntry {
    char filename[14];
    long size;
    time_t ftime;
};

struct direntry dent[MAX_DIRENTRIES];
int lastdent = 0;
int compf(struct direntry *d1, struct direntry *d2)
{
    return (int)( d1->size – d2->size );
}
/* čitaj dir i popuni dent sa lastdent elemenata */
qsort(dent, lastdent, sizeof( dent[0] ), compf);
..
```

Funkcija qsort podrazumeva sortiranje u rastućem nizu – sortiranje u opadajući niz se izvodi jednostavnim invertovanjem rezultata compf funkcije.

Interesantno je da qsort funkcija iz MS biblioteke ima ugrađenu i jednu proveru kako bi se izbegao granični slučaj koji ni malo ne prija qsort algoritmu – ukoliko primeti da je niz već sortiran, funkcija odmah završava rad.

SPECIJALNI KOMERCIJALNO-INFORMATIVNI BLOK
• akcije • marketing • publicitet •

Poziv za učešće u specijalnom dodatku

TEHNOTRONIKA '92.

Sajamski vodič — povodom sajma računarske opreme na 38. međunarodnom sajmu tehnike 11-15. maja 1992. u Beogradu

Povodom tradicionalnog sajma TEHNOTRONIKA redakcija „Računara" priprema, takođe tradicionalni SAJAMSKI VODIČ.

Mnogobrojne firme već su obezbedile mogućnost da u ovom specijalnom komercijalno-informativnom dodatku, zasebnom bloku na preko 50 strana, predstave i ponude hardver, softver, periferije, prateću opremu, usluge konsalting održavanje, obučavanje, poslovne ideje, stručna znanja i iskustvo. Sajamski vodič „Računara" će se i ovog puta potvrditi kao specifična i ugledna računarska berza, koja privlači veliku pažnju poslovnog sveta, kupaca, potrošača, korisnika i javnosti uopšte.

"Računari" se štampaju u najkvalitetnijoj tehnici (tabačni ofset), na najboljem papiru („kunstdruck" od 115 grama, korice 135 grama), sa ravnim rubom („rikna") Izvanredan „svetski" izgled „Računara", bogata sadržina i izuzetno kvalitetna reprodukcija oglasa svakako zaslužuju vašu pažnju.

Ako učestvujete na sajmu, svoje predstavljanje javnosti ćete, bez sumnje, pojačati svojim pojavljivanjem u našem sajamskom vodiču. A ako iz bilo kojih razloga niste prisutni na sajmu, verujemo da je za vas utoliko korisnija da učestvujete u našem vodiču. Jer, ovaj specijalni dodatak „Računara" će biti promovisan i prezentiran, prikazivan i reklamiran, gledan i korišćen, na takav način da je vaše učešće u njemu – jednostavno, pitanje prestiža.

Cene oglasnog prostora u sajamskom vodiču u „Računarima" iznose:

- 1/1 poslednja strana korica 300 000 D
- 1/1 druga strana korica 270 000 D
- 1/1 treća strana korica 270 000 D.
- 1/1 kolor unutrašnja strana 220 000 D.
- 1/2 kolor unutrašnja strana 140 000 D
- 1/1 crno-bela strana 180.000 D
- 1/2 crno-bela strana 120.000 D
- 1/4 crno-bela strana 80 000 D
- 1/8 crno-bela strana 60 000 D

Krajnji rok za rezervaciju prostora **20 april**

Krajnji rok za dostavljanje materijala za štampu **25. april.**

Za sve dogovore izvolite nas potražiti na

tel. 011/[illegible], fax [illegible], modem [illegible]
(osoba za kontakt: **Esad Jakupović**)

To još niste videli!

NA POKLON
486 • 386 • 386SX

U velikoj nagradnoj igri od sledećeg broja firma „ComTrad" vam poklanja tri moćna računara.

486
33 MHz, 4 MB RAM hard 105 MB super VGA, dva flopija tastatura, tower kućišta, miš

386
33 MHz, 4 MB RAM hard 105 MB super VGA, dva flopija tastatura tower kućište miš

386SX
20 MHz 1 MB RAM hard 40 MB, super VGA dva flopija tastatura baby tower kućište miš

i 30 utešnih nagrada

- 10 flopija 1,44 MB
- 10 miševa
- 10 kutija disketa

Ne propustite sledece „Računare" i svoju šansu da besplatno dobijete 486 — 386 — 386SX.

---

1) Turbo Pascal - naredbe i objekti

Nasleđivanje, virtuelni metodi, klase, objekti, opis svih naredbi, prevod svih grešaka, ilustracija DOS i BIOS poziva, opis direktiva prevodiocu. Svi primeri napisani su korišćenjem objekata! **Svet Kompjutera, januar 91:** "Po preciznosti i sveobuhvatnosti izlaganja, ova knjiga zamenjuje odgovarajuće delove originalne dokumentacije." **Računari, februar 1991:** "Pouzdan i iscrpan priručnik u svakodnevnom radu.." 240 strana, autor: Duško Savić.

2) Turbo Pascal 6.0 - turbovizija i grafika

Opis svih metoda, kompletno objašnjenje svih klasa i cele hijerarhije turbovizije. Programski primeri: editori, meniji, resursi, tokovi, dizajn interaktivne aplikacije... Izgradnja nove hijerarhije grafičkih objekata: mnogouglovi, matrice, linearne transformacije u 2D i 3D, perspektivna transformacija... Hiljade i hiljade naredbi! **Na ovo ste čekali!!!** 200 strana, autor: Duško Savić.

Knjige pod 1) i 2) su komplet za Turbo Pascal 6.0!

3) Uvod u objektno programiranje

Svet Kompjutera mart 92. "Primer studiozno organizovane i napisane knjige – Poglavlja su posvećena objektno orijentisanim programskim jezicima. Turbo Pascal 5.5, Turbo Pascal 6.0, TopSpeed Modula 2, C++ i Smalltalk/V .. **Ako želite objektno da programirate neka ovo bude vaša prva knjiga iz te oblasti.**", 245 strana, autor: Duško Savić.

4) Efikasne baze podataka

Prvi deo dBASE i efikasne baze podataka **za programera.** Drugi deo. JPI Modula 2 i efikasne baze podataka **za korisnika.** (Programi iz drugog dela izvršavaju se 100 150 puta brže od onih u prvom.) **Za sve jezike i programere!** stranična organizacija podataka nasuprot indeksnoj, kako upisati datum u dva bajta umesto u osam... **Profesionalci!** Pišite *efikasne baze podataka*, a mušterije umesto da kupuju brži hardver neka više plate *Vas*. Knjiga za svakog dBASE i Clipper programera! 200 strana autor: mr Dragoljub Savić.

Mogu se dobiti i diskete sa programima iz knjiga 1), 2) i 4)!

Cena po knjizi je dinarski ekvivalent od 10 DEM na dan uplate. Cena diskete po knjizi: 5 DEM. Isporuka pouzećem!

PC Program, Generala Hanrisa 20, 11040 Beograd, tel/fax (011) 463 296 žiro račun: 60816-603-36098

I dalje aktuelno: programi KUFI i KUIM - po 2700 din!

KOMPJUTEROM DO ZARADE obavljajući kompjuterske poslove! Tražimo saradnike. Za obaveštenja poslati adresirani koverat sa markom.
KLUB POSLOVNIH KOMPJUTERAŠA, Nenad Stojković, 21000 Novi Sad, Put partizanskih baza 8, tel. 021/387-743

MIX — Prevodioc (ENGLESKO-SRPSKI) PC program za obradu teksta sa ugrađenim rečnikom. Za obaveštenja poslati adresirani koverat sa markom.
KLUB POSLOVNIH KOMPJUTERAŠA, Nenad Stojković, 21000 Novi Sad, Put partizanskih baza 8, tel. 021/387-743

IBM PC & COMPATIBLE. Izrada i ponuda programa za PC računare iz svih oblasti.
NOVO — BESPLATNO: Najveći izbor PC software-a (Public Domain) u zemlji! Imamo **SVE! KATALOG.** Besplatv servisi inženjera, specijalista! EE Software, F. G. Marića 31 78000 Banja Luka, tel. 078/40-94[illegible].

NOVO!!! 8031/51 PC-razvojni sistem. Prodajem razvojni sistem i matične ploče za 8031/51 mikrokontroler tel. 018/719-773, 719-186

KOMPJUTER SERVIS
— SPEKTRUM FOLIJE
— COMODORE 64 i periferije
— AMIGA 500
— PC
Avalska 5 (na Slaviji) tel. 011/434-267

Vexcel notebook 386SX 16MHz 4 MB 40 HD, rezervna baterije brzi punjač. Upotrebljavan pouzdan 4 000 DEM Tel 011/664 309

POVOLJNO PRODAJEM DISKETE 5.25 3.5 /HD DD/ GARANCIJA. Tel /018/716-621/ 47 938

Džepni računari „Sharp" PC-E500 za inženjere i studente Tel 011/686-253

MN COMPUTERS
PC AT 286, 386
PRINTERI STAR, EPSON
MODEMI INTERNI, EXTERNI
tel. 011-444-6518

PRODAJEM ZVEZDICE ZA DŽOJSTIKE (PALICE) tel 030/83-203

- SOFTWARE — AMERIČKI ORIGINAL, 100% VIRUS FREE WINDOWS, OS/2, LOTUS, AMI
- DISKETE — DD/HD (USA) DOŽIVOTNA GARANCIJA
- PORTABLE DRIVES (USA) 0.9 kg 40/80/120 MB SVAKI PC i LAPTOP
- PERIFERIJE I PRIBOR 7—10 DANA OD UPLATE

N-SOFT
☎ 135-580
POSLE PODNE

REČNICI za PC

RESH 2.21+ Englesko-SH		NESH 1.10+ Nemačko-SH (pabe)
55.000	reči	38 000
1.5 MB	zauzeće diska	3.5 MB
3 550 [illegible]	din.	3 550 [illegible]

Mogućnost prevođenja i sa komandne DOS linije.

Clipper PROGRAMERI: Nudimo prevod uputstva za Clipper Debugger 5.01 (1 800 [illegible])

A-soft p.p. 81 31000 UŽICE tel 031/27-832



Priručnici iz pera iskusnih korisnika omogućiće vam da vrlo brzo ovladate korištenjem softvera i svu pažnju posvetite produktivnom radu sa računarom!

COMPUTER HIT BIBLIOTEKA
71000 Sarajevo Sime Milutinovića 4/II

NARUDŽBENICA

NARUČUJEM SLIJEDEĆE NASLOVE IZ EDICIJE "DŽEPNA KNJIGA ZA PC"

1	2	3	4	6	7	8	9	10	11	12	13	14	15	16	17	18

Ime i prezime (naziv firme) ____________

Poštanski broj i mjesto ____________

Ulica i broj ____________

Telefon telefax ____________

1	MS DOS 3.3/4.0	125 strana	1.125.00 din
2	PC Tools Deluxe 6.0	135 strana	1.125.00 din
3	WordPerfect 5.1	130 strana	1.125.00 din
4	MS Quick Basic	140 strana	1.125.00 din
6	Lotus 1-2-3 Rel 2.2/3.0	150 strana	1.125.00 din
7	Biblioteka Runtime funkcija za MS C	250 strana	1.625.00 din
8	MS Windows 3.0	155 strana	1.125.00 din
9	Clipper 5.0	310 strana	1.825.00 din
10	AutoCAD 10	280 strana	1.625.00 din
11	Ventura Publisher 3.0	230 strana	1.625.00 din
12	CorelDRAW! 2.01	160 strana	1.125.00 din
13	QuattroPro 2.0	170 strana	1.125.00 din
14	Word for Windows 1.1	160 strana	1.125.00 din
15	MS Excel 3.0	150 strana	1.125.00 din
16	Norton Utilities 5.0	120 strana	1.125.00 din
17	MS DOS 5	180 strana	1.125.00 din
18	Unix - uvod u rad	160 strana	u pripremi
19	Personalni računari - građa i primjena	300 strana	u pripremi
20	Programiranje u Windows okruženju	160 strana	u pripremi

Molimo da nas prije narudžbe kontaktirate u vezi cijene

Knjige možete naručiti poštom priloženom narudžbenicom ili telefonom na brojeve 071/270-213, 276-680, 621-025, fax 276-680 od 09 do 17 sati

INSTITUT ZA NUKLEARNE NAUKE "VINČA"
CENTAR ZA PERMANENTNO OBRAZOVANJE
11000 Beograd, Nemanjina 4/X
Telefoni: 011/ 683-390, 682-486, 641-155/ 107,181
Telefax: 011/ 682-486

KNJIGE U OBLASTI PROGRAMIRANJA, PROGRAMSKIH JEZIKA I PRIMENE RAČUNARA

1. AutoCAD (verzija 11.0)
konstruisanje i projektovanje pomoću personalnih računara
Autori: Boris Damjanović i Petar Damjanović
Sedmo izdanje: izlazi iz štampe u junu 92.

2. Uvod u C jezik
Autor: Vladan Vujičić
Četvrto izdanje 1991 - latinica 317 strana format B 5 broširan povez

3. Primena programa SYMPHONY na personalnim računarima
Autor: Dragan Pantić
Treće izdanje 1990 - latinica 226 strana format B 5 broširan povez

4. OS/2 - vodič za korisnike
Autor: Zorica Jelić
Prvo izdanje 1989 - latinica 253 strana, format B 5 broširan povez

5. VENTURA - računarsko izdavaštvo
Autor: Predrag Davidović
Treće izdanje: izlazi iz štampe u junu 92.

6. FORTRAN 77
standard sa dopunama za personalne računare
Autori: Vlajko Kocić i Zoran Konstantinović
Drugo izdanje 1990 - latinica 422 strane format B 5 broširan povez

7. UNIX - vodič za korisnike
Autor: Zorica Jelić
Treće izdanje 1991 - latinica 422 strane format B 5 broširan povez

8. Primena programa FRAMEWORK III na personalnim računarima
Autor: Dragan Pantić
Drugo izdanje 1991 - latinica 326 strana format B 5 broširan povez

9. Programski alati u matematici
MathCAD, GRAPHER, EUREKA
Autor: Ante Čurlin
Prvo izdanje 1990 - latinica 402 strana format B 5 broširan povez

10. Primena programa QUATTRO na personalnim računarima
Autor: Dragan Pantić
Prvo izdanje, 1992 - latinica 296 strana format B 5 broširan povez

11. DOS UKRATKO - verzije 3.3 i 5.0
Autor: Dragan Pantić
Drugo izdanje, 1992. - latinica 120 strana format B 5 broširan povez

12. Vodič za VAX/VMS
Autori: Tamaš Kerepeš, Zvonko Oršolić, Saša Marjanović
Prvo izdanje 1990. - latinica 512 strana format B 5 broširan povez

13. Primena programa EXCEL na personalnim računarima
Autor: Dragan Pantić
Prvo izdanje 1991 - latinica 272 strane, format B 5 broširan povez

14. UNIX - vodič za programere
Autor: Zorica Jelić
Prvo izdanje, 1991 - latinica 326 strana format B 5 broširan povez

15. WINDOWS 3.0
Autor: Dragan Pantić
Prvo izdanje 1991 - latinica 273 strane format B 5 broširan povez

16. PRIMAVERA
upravljanje projektima uz pomoć računara
Autori: Jaroslav Urošević i Jelica Drašković - Ostojić
Prvo izdanje 1991 - latinica 365 strana format B 5 broširan povez

17. dBASE III+ priručnik
Autor: Milorad Filipović
Prvo izdanje 1991 - latinica 249 strana format B 5 broširan povez

18. Osnovi informaciologije i informacione tehnologije
Autor: Ljubomir Dulović
Prvo izdanje 1991 - latinica 538 strana format B 5 broširan povez

19. LOTUS 1-2-3 (verzije 3.0 i 3.1)
Autor: Dragan Pantić
Prvo izdanje 1991 - latinica 411 strana format B 5 broširan povez

20. dBASE IV priručnik
Autor: Ljubomir Lazić
Prvo izdanje 1991 - latinica 306 strana format B 5 broširan povez

21. WORDPERFECT (verzija 5.1)
Autor: Dragan Pantić
Prvo izdanje 1991 - latinica 300 strana format B 5 broširan povez

22. Programiranje u CLIPPER-u 5.01
Autor: Alempije Veljović
Prvo izdanje 1992. - latinica 465 strana format B 5 broširan povez

23. FoxPro
Autor: Dušan Čalić
Prvo izdanje 1992. - latinica 390 strana format B 5 broširan povez

24. Uvod u strukture podataka
Autor: Miroslav Jocković
Prvo izdanje 1992. - latinica 357 strana format B 5 broširan povez

25. ORACLE (verzija 5.)
arhitektura i administracija
Autor: Vladimir Milojković
Prvo izdanje 1992. - latinica 151 strana format B 5 broširan povez

"RAČUNARI", april 1992

Poručujem (ispod rednog broja knjige upisati broj poručenih primeraka)

Redni broj knjige	1.	2.	3.	4.	5.	6.	7.	8.	9.	10	11.	12.	13.	14.	15.	16.	17.	18.	19.	20.	21.	22.	23.	24	25.
Broj poručenih primeraka																									

IME I PREZIME
(Naziv preduzeća) ____________________
ULICA I BROJ ____________________
BROJ POŠTE I MESTO ____________________ TELEFON ____________

Porudžbenicu poslati sa petim primerkom uplatnice (informacije o cenama svakog radnog dana od 8 do 20 sati) na adresu:
Institut za nuklearne nauke "Vinča"
Centar za permanentno obrazovanje, Beograd, Nemanjina 4/X.
Uplata se vrši na žiro račun
INSTITUT ZA HEMIJSKU DINAMIKU I PERMANENTNO OBRAZOVANJE, broj: 60803-603-17361
Troškove poreza na promet i isporuke knjiga snosi poručilac i namiruje ih unapred zajedno sa uplatom za knjige



qCAD

Program za projektovanje štampanih ploča za PC kompatibilne i ATARI ST računare

FUNKCIJE PROGRAMA qCAD:

Interaktivno kreiranje biblioteke oblika elemenata

Interaktivno unošenje signalne liste

Automatsko preuzimanje signalne liste iz drugih programa

Interaktivno raspoređivanje elemenata

Automatsko (100%) raspoređivanje elemenata

Interaktivno povezivanje signala pre i posle autorutera

Automatsko (100%) povezivanje signala sa optimizacijom

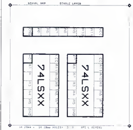
Automatsko kreiranje maski REPORT i NC-drill datoteke

Interaktivno editiranje i testiranje svih maski pre crtanja

Izlaz na matrične i laserske štampače, plotere i foto plotere.

Prikaz programa je dat u časopisu RAČUNARI 79 (februar 1992)

Program je autorsko delo Mr Čarapić Aleksandra iz Beograda



PERFORMANSE PROGRAMA qCAD:

Broj elemenata pinova i signala praktično neograničen

Veličina ploče do 800 mm sa 800 mm, broj slojeva od 1 do 6

Izbor veličine rastera i opcije automatskog umetanja džampera

Podrška običnih i SMD komponenata na rasteru od 1 mila

Sve performanse odnose se i na interaktivni i na automatski rad

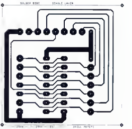
Ukupno vreme potrebno za razvoj ploča od 5 minuta do 1 časa

Ploča koju vidite na slici rezultat je automatskog projektovanja

Cena: 800 DEM (dinarski po kursu na slobodnom tržištu)

U cenu je uračunata godišnja garancija i zaštita od kopiranja

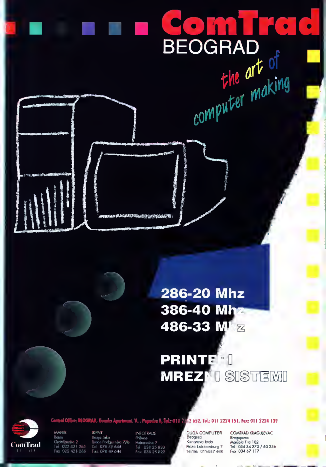
Garancija u vidu vracanja novca se ostvaruje u roku od 15 dana.



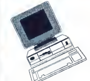
Microcad

Aleksandar Čarapić,
Lole Ribara 4/III, 11000 Beograd
Telefon: 011-346-432, 011-146-822


GAMA
Electronics
Gama Electronics, Mišarska 11, Beograd. Tel: 33 22 75; 33 94 94; fax: 335 902
Radno vreme: od ponedeljka do petka 9-17 h
Gama Electronics, München. Tel: 99/49/89/577 209; fax: 99/49/89/570 4379



ComTrad
BEOGRAD
the art of
computer making
286-20 Mhz
386-40 Mhz
486-33 Mhz
PRINTERI
MREZNI SISTEMI
Central Office: BEOGRAD, Tel.: 011 2224 151, Fax: 011 2224 139
MANIR
Ruma
Tel 022 421 265
Fax 022 421 265
IDENT
Banja Luka
Braca Podgornika 77b
Tel 078 49 644
Fax 078 49 644
INFOTRADE
Pristina
Tel 038 25 830
Fax 038 25 822
DUGA COMPUTER
Beograd
Karaburma brdo
Roze Luksemburg 7
Tel/fax 011/587 468
COMTRAD KRAGUJEVAC
Kragujevac
Marsala Tita 102
Tel 034 34 270 / 60 336
Fax 034 67 117
ComTrad



PONUDA MESECA

Interni streamer
COLORADO DJ10

120Mb

cena 750 DM



RAČUNARI
286, 386, 486
koprocesori
hard diskovi
monitori
ethernet
modemi i fax/modem kartice

Interni modem 2400 bauda 200DM

PC CLUB d.o.o.
Beograd, XXI divizije 44
tel. 444-30-79
fax. 422-199

ISPORUKA ! ODMAH

PROGRAMSKI PAKETI
DOS, XENIX, NOVELL
Programski jezici
Baze podataka
Obrada teksta
Obrada grafike
WINDOWS i aplikacije
Programi na CD ROM-u

MS DOS 5.0 170DM

REY
Beograd, Glasinačka 31/52
tel/fax 488-54-72

AKVIZICIJA PODATAKA, MERENJE I UPRAVLJANJE ZA PC RAČUNARE

Nabavili ste PC. Možete ga maksimalno iskoristiti ako na njega priključite jedan od naših sistema za akviziciju podataka, merenje i upravljanje. Na raspolaganju su Vam:

* ED1000 sistem sa 256 ili 512 analognih ili digitalnih kanala,
* ED2000, koji se instalira u PC, popularan model sa srednjom cenom i
* ED300-MONITOR, mali sistem koji ima minimalnu (tajvansku) cenu.

Ovi uređaji se koriste za akviziciju podataka, procesno upravljanje u realnom vremenu, beleženje podataka (Data Logger), automatsko testiranje kvaliteta proizvoda itd.

Softver: LABTECH NOTEBOOK, DADISP, ASYST, RELAY LADER LOGIC, ONSPEC, LABTECH CONTROL, domaći MIS itd. Ukoliko želite da napišete svoj softver na raspoloženju su Vam softverski drajveri za BEJZIK, C, i PASKAL.

Tražite da vam pošaljemo našu novu knjigu-katalog sa preko 200 stranica teksta: AKVIZICIJA PODATAKA, MERENJE I UPRAVLJANJE NA PC RAČUNARIMA.

ED ELEKTRONIK DIZAJN

M. Tolbuhina b.b. — Pejton
11000 Beograd
Tel. (011) 450 480,
Fax. (011) 444-7459
Radno vreme je od 9 do 18 časova




EPSON
VELIKO
IME,
POUZDAN
KVALITET!
EPSON
EPSON
COMPUTERS

isporuka SIGURNA **IMTEL U BEOGRADU** garancija 2 godine



MONITOR / HARD DISK / RAČUNAR	MONOHROMATSKI VGA			KOLOR VGA		
	40MB	105 MB	210 MB	40MB	105 MB	210MB
486-33, 4MB RAMA, MINITOWER	4510	5180	5880	5030	5830	6530
386-40, 2MB RAMA, MINITOWER	3810	4480	5180	4480	5130	5830
386SX-25,2MB RAMA, STAND KUĆIŠTE	2680	3350	4050	3330	4000	4700
286-16, 1MB RAMA, STAND KUĆIŠTE	1880	2550	3250	2530	3200	3900

CENE SU U DM
I OBRAČUNAVAJU SE U DINARE PO KURSU MARKE NA SLOBODNOM TRŽIŠTU

ŠTAMPAČI	
HP LASER JET III P	3700
MEM. PROŠ. ZA LJ III P	400
TONER ZA LJ IIP	330
EPSON FX 1050	1450
EPSON LX 800	700

KOPROCESORI	
AMD 80C287-12	180
80387SX-20	550

MODEMI 2400 BAUDA.	
POCKET	200
EKSTERNI MNP5	400

SOFTVER:	
DR DOS 4.3	100
MS DOS 5.0	250
MS WINDOWS 3.0	350

EPROM PROGRAMER	400
HP IB KARTICA	550
AD/DA KARTICA	950
STRIMER COLORADO 120MB	800
MIŠ GM-6	55
FLOPI 1.44MB	150

DISKETE:
1.2 MB
1.44MB

NOVELL NETWARE
MREŽNI OPERATIVNI SITEM

intel inside

RAČUNARI SU JAČI KADA SU POVEZANI !



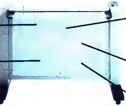
NOVELL NETWARE 386 V3.11 PODRŽAVA DO 100 KORISNIKA	16000 DM
NOVELL NETWARE 386 V3.11 PODRŽAVA DO 20 KORISNIKA	8000 DM
NOVELL NETWARE 286 V2.2 PODRŽAVA DO 50 KORISNIKA	8000 DM
NOVELL NETWARE 286 V2.2 PODRŽAVA DO 10 KORISNIKA	5000 DM
ETHERNET MREŽNA KARTICA NE2000	550 DM

IMTEL - Institut za Mikrotalasnu Tehniku i Elektroniku

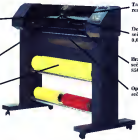
Bulevar Lenjina 165 b, 11070 Novi Beograd

Telefon: (011) 134-516, Fax: (011) 138-928, Telex: 12797 IMTELYU, Žiro račun 60815-603-14396

Skeneri



Sekači



Vrhunske grafičke periferije, softver i inženjering

NOVO! NOVO! FAMILIJA PLOTERA EDMP 160 sa još boljim performansama

kma D.O.O.
11070 N. Beograd
Milentija Popovića 9 (Sava centar)

Tel 011/222-19-28
Tel 011/439-618
Tel 011/435-715
Tel 011/222-43-22, lokal 453
Fax 011/455-785 605-578

Ovlašćene prodajne organizacije:

SoftWell d.o.o.
Kompjuterski inženjering
Pozorišni rg 1
21000 Novi Sad
Tel 021/51-999 52-396

El „Računari" – Niš
Bul. V. Vlahovića 80–84
18000 Niš
Tel 011/325-461 55-583
Fax 325-723

Laneco
Braća Pavlića H
78000 Banjaluka
Tel 078/33-992 48-717
Fax 33-992

Makpetrol kompjuteri
Kompjuterski inženjering
Ul. „Orce Nikolov" bb
91000 Skopje
Tel 011/229-676 221-040
Fax 229-676

µSOFT

Drinčićeva 20, tel. 320-776

OVLAŠĆENI DISTRIBUTER KORPORACIJE **MICROSOFT**



Da biste pronašli blago, ne morate da ga tražite SA ONE STRANE DUGE. Blago se nalazi kod NAS.

Kupite LICENCIRANE pakete :

WINDOWS 3.0 , 3.1	QUICK PASCAL COMP. 1.0
WORD 2.0 FOR WIN	BASIC PDS 7.1
EXCEL 3.0 FOR WIN	C PDS 6.0
WORKS 2.0 FOR WIN	MACRO ASSEMBLER 6.0
WORD 5.5	FORTRAN 5.1
QUICK C FOR WIN	COBOL PDS 4.5



I MNOGE DRUGE

CET . OVLAŠĆENI DILER . TEL. 343 - 043

SOFT LAND

OVLAŠĆENI SUBDISTRIBUTER KORPORACIJE **BORLAND**

SKADARSKA 45. TEL. 848-5698



TURBO C++ FOR WIN	PARADOX 3.5
TURBO PASCAL FOR WIN	QUATRO PRO 3.0
C++ & APP FRAMEWORKS 3.0	dBASEIV
TURBO PASCAL 6.0 PROFF	TURBO PASCAL 6.0
C++ 3.0	TURBO C++

TURBO C++ & APPLICATION FRAMEWORKS 3.0

i drugi

DŽON FLINT, EN BONI, RIĐOBRADI, KAPETAN KID - otišli su u istoriju. PIRATI su odigrali svoju ulogu. VI NE MOŽETE da sledite njihovu tradiciju! Kupite LICENCIRANE pakete !

CET . OVLAŠĆENI DILER . TEL. 343 - 043

MOST, Beograd, Kralja Bodina 2, tel 011/422-170, fax. 011/426-144

GARANCIJA 1 GODINA
ISPORUKA ODMAH
DINARSKO PLAĆANJE PO DNEVNOM KURSU DEM

BRUG Computers HOLLAND | **BRUCKE Computers** GERMANY | **MOST Computers** SRBIJA

CENE RAČUNARA

PC AT 286/16MHz.................................1.490 DEM
PC AT 386/25MHz.................................2.090 DEM
PC AT 386/33MHz/cache 64kB.........2.390 DEM
PC AT 386/40MHz/cache 64kB.........2.490 DEM
PC AT 486/33MHz/cache 64kB.........3.590 DEM

OSNOVNE KONFIGURACIJE

SVI RAČUNARI: FLOPPY OISK 5,25"/1,2MB ILI 3,5"/1,44MB, TVROI OISK 52MB IOE, "HERKULES GRAFIČKI AOAPTER MONOHROMATSKI PAPER WNITE MONITOR, TASTATURA, INSTALIRAN MS OOS 4.01, UPUTSTVO NA SRPSKOM
286 I 386 RAČUNARI: 1MB ILI 2MB RAM, MINI TOWER KUČIŠTE
486 RAČUNARI: 4MB RAM, TOWER KUČIŠTE

52 MB HDD

DOPLATE ZA VEĆE ZAHTEVE

VGA 1024X768 "HYUNDAI" KOLOR MONITOR,
"TRIDENT" TVGA9000 GRAFIČKA KARTA..............590DEM
TVRDI DISK 130 MB IDE 500DEM
TVRDI DISK 200 MB 900DEM
FLOPPY DISK 1,44 MB......................................159DEM
1MB 125DEM

CENE ŠTAMPAČA

"EPSON" LX-850, 9-PINSKI, A4.....................................690DEM
"EPSON" LQ-570, 24-PINSKI, A4...................................790DEM
"EPSON" LQ-1070, 24-PINSKI, A3.......................1.295DEM
"HEWLETT PACKARD" HP LASERJET IIIP, 1MB........................ zvati
"HEWLETT PACKARD" HP LASERJET III, 1MB.......................... zvati

MOST design: Snežana Vujović

ZVANIČNI DISTRIBUTER RAČUNARA "MOST" U SAVA CENTRU TEL.222-43-22/482

"MOST" RAČUN
"NEWLETT PACKARD" LASERSKI ŠTAMP
"EPSON" ŠTAMP
"NOVEL" RAČUNARSKE M
"MICROSOFT" OPERATIVNI SIST
POSLOVNI I GRAFIČKI SIST

POSETITE NAS NA SAJMU TEHNIKE U BEOGRADU OD 11-15.

NAJPOVOLJNIJE CENE NAJVIŠI KVALITET


POVEZIVANJE U MREŽE POD NOVELL NETWARE-OM, INDUSTRIJSKE I LAB-KARTICE "ADVANTECH"
I OVLAŠĆENA PRODAJA FIRME OTC COMPUTER, TAIWAN
ComTec
SVE STANDARDNE KONFIGURACIJE 286-386-486
LAPTOP
NOTEBOOK
PREDUZEĆE ZA INFORMATIČKI INŽENJERING I AUTOMATIZACIJU
NOVI SAD, VOJVODANSKIH BRIGADA 7/IX, Fax 021 / 222-58

OTC COMPUTERS

MILENTIJA POPOVIĆA 9
SAVA CENTAR
11070 NOVI BEOGRAD
YUGOSLAVIA
TEL 38/11/222 23 55
TELEX 12042 YU SAVCEN
FAX 38/11/222 23 55
TEL 38/11/136 840 EXT 684
TEL 38/11/222 43 22 EXT 684
TEL 38/11/222 43 23 EXT 684

I VI IMATE PRILIKU DA POSTANETE PLAĆENI AGENT INOSTRANE ORGANIZACIJE!

KULTURA ISTOKA TRAJE BEZ PREKIDA VEĆ PET HILJADA GODINA. REZULTAT TOGA JE I PRIVREDNI BUM TAJVANA. "OTC" JE PRVA TAJVANSKO-JUGOSLOVENSKA KOMPANIJA. BAVIMO SE PROIZVODNJOM I PRODAJOM PC RAČUNARA. MATIČNA KOMPANIJA SE NALAZI NA TAJVANU, A SESTRINSKE U BARSELONI, RAVENI, EDINBURGU, BUDIMPEŠTI, I POZNANJU. NUDIMO VAM DA BUDETE NAŠ PARTNER. AKO IMATE OSEĆAJ ZA DOBAR POSAO, ŽELJU DA BUDETE USPEŠNI I PRODAJNI PROSTOR, JAVITE NAM SE.





COMPAQ

It simply works better.

COMPAQ RAČUNARI SADA PO ZNATNO POVOLJNIJIM CENAMA

Ovlašćeni distributer

MS MICROSYS

Novi Sad – Beočin
Tel: (021) 611-366, fax: (021) 20-544

Poslovnica Beograd, Dobračina 39
Tel: (011) 625-994, fax: (011) 622-581

Jednostavno, mi smo pouzdani

ERC commerc

SIRMIUM COMMERC

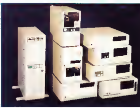
Kolarčeva 4/IV
Beograd
Tel 011/632-532
011/626-792
Fax 011/633-059

HERMANN PUCHMANN
IM-EXPORT
HAMBURG Germany

MEŠOVITO PREDUZEĆE EXPORT-IMPORT

STUDIOTECHNIK PUCHMANN D-2000 HAMBURG 70

KONFIGURACIJA JE VAŠ – A KVALITET NAŠ POSAO!



Modeli 286, 386 i 486
u svim mogućim varijantama.

Specijalno za vas softver
(za export-import firme,
agencije, zanatske radnje)
MS-DOS, Novell

Garancija 2 godine.

Isporuka sa lagera odmah
po uplati.

Izaberite komponente sa spiska — dobićete sklopljen i konfigurisan računar.

KUĆIŠTA	DEM
MODEL 3200 MD	188 –
MODEL 402	176 –
MODEL 402 D	184 –
MODEL 217	175 –
MODEL 501	225 –
MODEL 272	242 –
MODEL 486 BIG TOWER	390 –
DISK JEDINICE	
TEAC 1.44 MB	170 –
NEC 1.2 MB, 1.44 MB	170 –
MITSUMY 1.2 MB, 1.44 MB	170 –
MITSUBISHI 1.2 MB, 1.44 MB	170 –
TVRDI DISKOVI	
Western Digital	
40 MB 17 ms	513 –
80 MB 17 ms	827 –
120 MB 17 ms	991 –
Fujitsu	
45MB 24 ms	520 –
Quantum	
40 MB 17 ms	521 –
80 MB 15 ms	835 –
120 MB 15 ms	1001 –
TASTATURE	
CHICONY YU-1	99 –
CHERRY	204 –
ŠTAMPAČI	
STAR LC24–1, 24 iglični, A3 format	1320 –
NEC P 30, 24-iglični, A3 format	1350 –

MATIČNE PLOČE	DEM
286/16 MHz	195 –
286/25 MHz	235 –
386 SX-16	400 –
386 SX-25	491 –
386 DX-33, CACHE 64 KB	697 –
386 DX-33, CACHE 128 KB	940 –
486 DX-33, CACHE 128 KB	1700 –
D RAM 41256 8	3.70 –
D RAM 44256 8	14 –
SIMM 1MB××9–7	96 –
SIMM 256k×9–7	28.50 –
KONTROLERI	
HERCULES, EPROM Socke	28 –
AT BUS	52 –
AT BUS, 2×ser.+par.+game	62 –
VGA 800×600	98 –
VGA 1024×768	170 –
MIŠ GENIUS MOUSE GM 6	58 –
MONITORI	
MONOCHROM 14"	232 –
MONO VGA 1024×768	310 –
SVGA COLOR 14" 1024×768	832 –
NOVELL ETHERNET KARTE	
Novell 1000	422 –
Novell 2000	452 –
DISKETE	
Fuji 5.25" 1.2 MB	2.30 –
Fuji 3.5" 1.44 MB	4 –

Niko nije dovoljno bogat

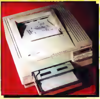


LaserJet IIIp

LaserJet III

LaserJet III D

LaserJet III Si



PaintJet

DeskJet 500c

ScanJet IIc

DraftPro



DesignJet

DraftMaster

HP 95 LX

da kupuje jeftine stvari

Izaberite kvalitet!

MICRO

MICRO AIR
11000 BEOGRAD, Kneza Miloša 17
Save Kovačevića 23/VI/27
Tel/Fax 011 453 352

HEWLETT PACKARD
Authorized Dealer